(1)

DEPARTMENT OF THE AIR FORCE

AIR UNIVERSITY

AIR FORCE INSTITUTE OF TECHNOLOGY

Wright-Patterson Air Force Base, Ohio

AFIT/GOR/ENS/91-M

(1)

LOCATING DIRECTION FINDERS IN A GENERALIZED SEARCH AND RESCUE NETWORK

THESIS

Jean M. Steppe
Captain, USAF

AFIT/GOR/ENS/91-M-17

AFIT/GOR/ENS/91-M

LOCATING DIRECTION FINDERS IN A GENERALIZED SEARCH AND RESCUE NETWORK

THESIS

Presented to the Faculty of the School of Engineering
of the Air Force Institute of Technology
Air University
In Partial Fulfillment of the
Requirements for the Degree of
Master of Science Operations Research

Jean M. Steppe, B.S.
Captain, USAF

March, 1991

THESIS APPROVAL

STUDENT: **Captain Jean M. Steppe** **CLASS:** **GOR 91-M**

THESIS TITLE: **Locating Direction Finders in a Generalized Search and Rescue Network**

DEFENSE DATE: **20 FEB 91**

COMMITTEE:	NAME/DEPARTMENT	SIGNATURE
Advisor	Dr. Yupo Chan/ENS	Yupo Chan
Co-advisor	Dr. James W. Chrissis/ENS	

Preface

The purpose of this research was to develop a mathematical location/allocation model for the generalized search and rescue problem (GSARP). The model proposed herein provides good feasible solutions for a single time block. Future research may show that this model formulation can be adapted to provide good feasible search and rescue network configurations for the multi-time period GSARP.

In the process of producing this thesis, I have had a great deal of help from others. I would like to thank my thesis advisors Dr. Yupo Chan and Dr. James Chrissis for their continuing insight and perspective. I would also like to thank my research sponsors Dr. Alfred Marsh, Capt. David Drake, and Ms. Sara Cohen for their assistance and technical support on this challenging problem. A word of thanks is also in order for Mr. Douglas Burkholder and Mr. Jack Phillips whose technical assistance with the computer services was invaluable. Finally, I wish to thank my family whose love and support in the final weeks carried me through to completion.

Jean M. Steppe

Table of Contents

Page

Page

List of Figures

List of Tables

AFIT/GOR/ENS/91-M

Abstract

A multiobjective linear programming approach is applied to the problem of locating receiving stations and HFDF receivers in a search and rescue network in order to maximize the expected number of distress signals that are geolocated. The multiobjective formulation is made up of two contrasting objectives: one maximizes the expected accurate lines of bearing, and one minimizes the excess coverage in the network. The individual objectives are weighted and combined into a composite objective function. The resulting problem is expressed as a two-stage network flow problem and is solved using SAS LP with a limited number of binary variables. The problem is iteratively solved for several weightings of the composite objective function. Solutions are evaluated by a FORTRAN program provided by the Department of Defense. In all cases, the best results were three to four standard deviations better than a sample of 1000 or more heuristically tasked random network configurations. These results demonstrate that a two-stage multiobjective formulation consistently provides good feasible network configurations and is, therefore, a practical alternative to the robust, yet intractable, nonlinear integer formulation.

LOCATING DIRECTION FINDERS IN A GENERALIZED SEARCH AND RESCUE NETWORK

I. Introduction

1.1 Background

(Cont)

The United States is building a worldwide network of search and rescue (SAR) stations for performing SAR over broad ocean areas. The objective of this network is to geolocate distress signals from aircraft and ships in order to initiate SAR missions. "An optimal SAR network design would maximize the expected successful geolocations delivered by the worldwide net over the ocean areas considered" (17:1). The design of an optimal SAR network entails allocating a limited number of receiving stations to a larger set of candidate receiving station locations and allocating a limited number of high frequency direction finder (HFDF) receiver assets among the selected receiving stations (17:2).

Receiving stations in the SAR network have a signal reception system which performs acquisition and direction finding (performed by HFDF receivers) to provide lines of bearing (LOB) for a distress signal (17:1). Drake describes the process of geolocating a distress signal (11:1–8). The process begins when Central Control is notified of the acquisition of a distress signal. Upon notification, Central Control requests operators at other SAR network stations to transmit LOBs for the signal of interest. To decrease the likelihood of grossly inaccurate position estimates, LOBs from at least three receiving stations are required before making a geolocational point estimate of the distress signal.

The SAR facility location and frequency allocation problem is currently handled with a brute force greedy approach in which each candidate station is evaluated with a fixed network of active stations to determine which combination of stations delivers the largest number of expected geolocations (5). Complete enumeration and comparison of all possible network configurations is a combinatorially explosive task and is not considered an efficient methodology for optimizing the expected number of geolocations in a SAR network.

1.2 HFDF Assignment Problem

The GSARP is related to the HFDF assignment problem researched by Drake and Johnson which attempts to maximize expected geolocations for a fixed network (11, 14). The GSARP and the HFDF assignment problem are similar in three respects. First, both problems seek to maximize the number of expected geolocations. Second, both problems must optimally assign frequencies to a limited number of HFDF receiver assets. Third, both problems must consider stochastic transmission and propagation probabilities.

Drake proposes a nonlinear objective function to optimize the expected geolocations (11). Drake's nonlinear function, while providing a robust description of the true objective and a means to measure how good a solution is, becomes computationally impractical to optimize as the size of the problem increases towards realistic sizes. For computational tractability, Johnson formulated the problem as a multiobjective linear integer program (MOLIP) using three objective functions which respectively maximized the expected number of lines of bearings, maximized expected number of transmissions, and minimized excess coverage of a frequency (14). Johnson solved the MOLIP using a network flow model. The best solution was more than 13 standard deviations better than the sample mean of a completely randomized set of HFDF assignments (14:37).

There are two additional degrees of difficulty in the GSARP that were not addressed by the previous research. The first is that the GSARP must determine optimal locations for additional receiving stations to be placed among a fixed base network of receiving stations, whereas the HFDF assignment problem dealt only with a fixed base network of stations. The second and most significant degree of difficulty is that the GSARP must locate HFDF resources in bundles of eight to the receiving stations, whereas the HFDF assignment problem used a predefined allocation to successfully solve the MOLIP using a network flow model (14).

1.3 Research Objective

The purpose of this research is to develop a mathematical programming model to locate direction finders in a generalized search and rescue (GSAR) network. Specifically,

the GSAR problem (GSARP) entails locating additional receiving stations with a fixed base network of receiving stations and assigning HFDF receivers and their frequencies to selected receiving stations so that the direction finders are in the best configuration to maximize the number of expected successful geolocations in a given time block for a SAR network.

1.4 Overview of Research Effort

The research effort presented in the remaining chapters includes a literature review, model formulation, computational experience with model formulations, solution methodologies, network representations and results.

Selected facility location problems are discussed in the literature review in Chapter 2. Chapter 3, Model Formulation, presents a nonlinear integer programming (NLIP) formulation developed from Drake's nonlinear formulation of the HFDF assignment problem, as well as a simplified multiobjective linear integer programming (MOLIP) formulation that was motivated by Johnson's HFDF assignment problem research (11, 14). Computational experience with these formulations is documented in Chapter 4 to support the use of the MOLIP formulation for the GSARP. Chapter 5 describes solution methodologies that are used to solve the GSARP. Linear and integer programming, multiobjective optimization strategies as well as motivation for network representations are all discussed in detail. Chapter 6 then discusses the potential of two specific network representations for the MOLIP formulation. The network representations are described with test case results validating the selection of a two-stage network representation.

Chapter 7, Results, presents thesis research results for the two-stage network representation of the MOLIP formulation. Results both with and without a covering constraint are presented for comparison. A DOD computer program called EVAL evaluates the approximate nonlinear objective function value for each MOLIP solution (8). The best solution for each time block with and without a covering constraint is compared with the mean and standard deviation of EVAL results found by the Department of Defense for a large set of randomly generated network configurations (each configuration satisfies the MOLIP constraints) that are each tasked by a greedy heuristic which maximizes the lines

of bearing at each individual station. This heuristic provides good feasible solutions for a fixed network (10).

This research shows that the two-stage MOLIP methodology consistently selects good feasible location configurations that are better than nearly all of the randomly selected network configurations which were heuristically tasked. The MOLIP heuristic is a robust methodology for locating HFDFs in a SAR network. It is effective compared to a random selection and efficient compared to the computationally intractable NLIP formulation.

II. Literature Review

2.1 Introduction

Selected facility location literature related to the generalized search and rescue problem (GSARP) will be presented. Specifically, this literature review covers location problems with uncertainty, maximum-expected-covering location problems, backup-coverage location problems, and equity-maximizing location problems. Literature concerning HFDF bearing accuracy and model selection will be covered as part of model formulation in Chapter 3. Literature related to solution strategy will be covered as part of solution methodology in Chapter 5.

2.2 Facility Location Problems

In "An Overview of Representative Problems in Location Research," Brandeau and Chiu give a general definition of a location problem (3):

> A location problem is a spatial resource allocation problem. In the general location paradigm, one or more service facilities ("servers") serve a spatially distributed set of demands ("customers"). (3:646)

The objective function of a typical facility location problem optimizes costs, such as distances or time, related to facility-facility or facility-demand interactions (3:646-647). In other words, facilities are located so that customer demand can be satisfied economically.

Two classes of facility location problems found in the literature are the *set covering* problem and the *p-median* problem. The set-covering problem minimizes the number of facilities located for a predetermined level of coverage, whereas the p-median problem requires that p facilities be located with the objective of being proximal to the demands on the average.

Facility location problems can have either uncapacitated or capacitated demand. The uncapacitated problem has no restriction on the demand that can be satisfied by each facility, while the capacitated problem restricts the amount of demand satisfied by each facility (19:7-8). The capacitated facility cannot always respond to all demands.

The GSARP can be modeled as a capacitated p-median facility location problem with the receiving stations acting as service facilities, the frequency bands at transmitter locations acting as customers, and the joint probabilities of transmission, propagation, and bearing accuracy acting as a weighted demand. The weighted demand acts like a fixed charge on the allocation of a specific frequency to a specific receiving station.

The following paragraphs discuss location problems with uncertainty and three types of capacitated facility location problems: maximum-expected-coverage location problems, backup-coverage location problems (models I, II, and III), and equity-maximizing location problems.

2.3 Location Problems with Uncertainty

Demand is sometimes uncertain when facility location decisions are made. In the GSARP, the transmission of a signal on a given frequency is given as a probability distribution for each distress location.

A two-stage stochastic model for production and location under demand uncertainties is considered by Louveaux and Thisse (16:145-149). During the first stage, the firm uses the predicted demand to choose the location and production capacity to maximize its expected profit utility. During the second stage, the firm uses the true demand to choose a production distribution schedule so as to maximize profit, given location and production decisions made in stage one.

The GSARP can be thought of as the first stage of optimization where the location of receiving station and HFDF assets is determined using stochastic demand. During the second stage of a two-stage stochastic model, the HFDF assignment problem could be solved using updated transmission and propagation probabilities.

2.4 Maximum-Expected-Covering Location Problem

Daskin considers the maximum-expected-covering-location problem where demand and server availability are unknown (7:48-68). This model specifically addresses the possibility that facilities may not be available to respond to demand. Daskin makes three

important assumptions: the probability p that a facility cannot respond to demand is the same for all facilities; the ability of a facility to be busy is independent of all other facilities being busy; and busy probabilities are invariant with respect to the server location (1:278). However, Daskin's assumption of equal and invariant busy probabilities does not hold for the GSARP. The GSARP has propagation probabilities representing the busy probability of an HFDF asset. These busy probabilities differ for each time period and for every combination of distress location, frequency, and receiving station.

Batta, Dolan, and Krishnamurthy reexamine Daskin's formulation of the maximum-expected-covering location problem (1:277-286). They investigate relaxing the three assumptions made in Daskin's model. In relaxing the assumptions, Batta, Dolan, and Krishnamurthy use a hypercube queuing model in a single node substitution heuristic to determine the set of server locations which maximizes the expected coverage (1:277-286). They adjust Daskin's maximum-expected-covering location problem, based upon random sampling of servers without replacement, to produce results in better agreement with the hypercube queueing model. Batta, Dolan, and Krishnamurthy conclude that all three models (Daskin's maximum-expected-covering location problem, hypercube queueing, and adjustment of Daskin's maximum-expected-covering location problem) produce results of similar quality (recommended facility locations are physically close to one another). However, it is noted that computational intensity and accuracy of coverage estimation differ for each model (1:278).

The GSARP can be conceived as a maximum-expected-covering location problem. The demand is stochastic, represented as a probability distribution for each distress signal location for a given time period. Server availability is also stochastic, represented as the probability that a signal propagates on a particular frequency from a distress signal location to a receiving station location for a given time period.

2.5 Backup Coverage

Backup coverage involves the second and subsequent coverage of a demand node by service facilities. The efficient coverage of stochastic demand and server availability, may require backup coverage in areas of high demand (13:1434). The GSARP which has

stochastic demand and server availability requires frequencies be covered by at least three HFDF assets for geolocations to be attempted on that frequency. Additional HFDF assets in the network will result in backup coverage. The GSARP reduces to locating receiving stations and HFDF assets and allocating backup coverage among the assets to maximize the number of geolocations in the network. The following paragraphs review the Hogan and Revelle and the Pirkul and Schilling concepts of backup coverage (13, 21).

2.5.1 Backup Coverage Model I (BACOP1). Two multiobjective models for backup coverage are presented by Hogan and Revelle (13:1437-1440). Their first model, BACOP1, incorporates aspects of both the set-covering location problem and the maximum-covering location problem. BACOP1 has two objectives. The first objective is to minimize the number of facilities sited which ensure primary coverage of demand. The second objective maximizes the amount of demand that is provided backup coverage. In this formulation, a structural backup coverage variable is assigned the value of one to identify when a particular demand is covered two or more times and is assigned the value of zero otherwise. The backup coverage variable for a particular demand node is weighted by the amount of demand at that node. This insures that some level of backup coverage is provided for the maximum amount of demand. This model can also be extended to levels of coverage greater than two with additional constraints and an additional objective for each level (13:1443).

The GSARP has some similarities and differences to the backup coverage model BACOP1 (13:1437-1440). Both require primary coverage of demand. However, BACOP1 is a set-covering problem minimizing the number of facilities needed for primary coverage of all demands. The GSARP, on the other hand, is a p-median (p-facility) maximal coverage problem which can be constrained to provide primary coverage of three HFDF's per frequency.

2.5.2 Backup Coverage Model II (BACOP2). Hogan and Revelle's second backup coverage model, BACOP2, permits assignment of backup coverage to areas of high demand, prior to assigning first coverage to areas of low demand (13:1437-1441). The BACOP2 formulation, an extension of the maximum covering location problem, is multiobjective.

The first objective maximizes the demand that receives primary coverage, and the second objective maximizes the demand that receives backup coverage. The amount of demand receiving primary and backup coverage is maximized because the structural variables in the objective functions representing a given demand node are weighted by the amount of demand at that node.

The backup coverage model BACOP2 discussed by Hogan and Revelle has two significant crossovers to the GSARP (13:1437-1441). First the GSARP requires a p-median maximal covering model which BACOP2 represents. Second, the concept of weighting both primary and backup coverage by the demand for coverage is a reasonable way to model the GSARP.

2.5.3 Backup Coverage Model III. Pirkul and Schilling discuss a maximum-covering location problem where primary and backup services are required from separate facilities for each demand [illegible] 140-153). This model is applicable when considering demand for emergency services. In this situation, it is desirable to have backup coverage available within a certain distance if primary coverage is not immediately available. Pirkul and Schilling's research recognizes the negative effect of assigning demand to a facility that is a great distance away. Their model attempts to provide acceptable assignments for uncovered demands by using a declining distant-dependent function for demand assignments that exceed the acceptable coverage distance.

Pirkul and Schilling's distant-dependent function has potential crossovers to the GSARP. For the GSARP a similar concept, called an accuracy-dependent function could be used to weight each line of bearing (LOB). This type of a weighting function could embody the negative effect of assigning demand to a facility that has large LOB errors.

2.6 Equity Maximizing Location Problems

Most facility location problems maximize benefits or minimize cost (or travel distance) across all demand in the system. This is efficient but not equitable, because some customers must travel farther to receive the same benefit (2:137). This inequity becomes relevant when all customers pay the same fee to use a facility. In this case, all customers

should receive the same benefit. Berman and Kaplan consider a facility location problem which attempts to equalize facility benefits for all customers by using taxes or side benefits (2:137-138). Their formulation attempts to maximize the benefit derived per customer by using an approach which minimizes the sum of absolute taxes or side payments that would be required for a given set of facilities and demands to be in equilibrium (2:140-143).

The GSARP might benefit from equity-maximizing considerations. For the HFDF assignment problem, Johnson discusses the decreasing utility of assigning additional HFDF receivers to frequencies once a frequency is adequately covered (14:17-18). To deal with this, Johnson used an objective which penalizes assignment of additional HFDFs to a frequency that already has its *fair share* of HFDF resources (14:18-19). If there were no penalty for excess coverage, HFDF assets would be assigned to frequencies with the greatest probabilities of transmission and propagation with no regard for the number currently assig. . ¹ to the frequency. This objective imposes a side payment to the assignment of HFDF resources which exceed the equal coverage criteria.

2.7 Conclusions

The GSARP is best characterized as a capacitated maximum-covering location problem. The problem is capacitated due to the limited number of HFDF receiver assets available to fulfill demand. The GSARP is maximum covering, since the objective is to maximize the number of geolocations.

Hogan and Revelle discuss a backup coverage model for a maximum covering location problem, BACOP2, which has potential crossovers to the SAR problem (13:1436-1437). Two modifications can make the BACOP2 formulation more suitable for the GSARP. First, primary coverage for all significant demands should be required. Second, each unit of primary and backup coverage should be weighted by the amount of demand it is expected to satisfy.

Pirkul and Schilling present the concept of a declining distant-dependent function to incorporate the negative effect of assigning a facility to a demand that is far away. For the GSARP, a similar concept of a declining accuracy-dependent function could be used

to weight each line of bearing. This weight could embody the negative effect of assigning demand to a HFDF asset to a station that has large LOB errors.

An equity-maximizing objective could also be used for the GSARP, similar to the way Johnson penalized excess coverage for the HFDF assignment problem (14:17-18). This objective recognizes the decreasing utility of assigning additional HFDF receivers to a frequency that is already adequately covered.

III. Model Formulation

This chapter presents both the nonlinear and the linear multiobjective formulations of the generalized search and rescue problem (GSARP), along with a summary of the literature motivating the multiobjective linear formulation. The first section describes the scoped thesis problem, as well as the comprehensive GSARP. This is followed by a complete description of the notation used for the nonlinear and linear models. The last two sections present the nonlinear and the multiobjective linear formulations.

3.1 The Generalized Search and Rescue Problem (GSARP)

The GSARP requires that HFDF receiving stations and receiver assets be located within a fixed search and rescue (SAR) network so that the number of expected geolocations is maximized. Each *receiving station* location has a set of *propagation* probabilities which are unique for each combination of time block, transmitter, station, and frequency. Each *transmitter* location has a set of *transmission* probabilities which are unique for each combination of time block, transmitter, and frequency. A 24 hour day is broken into 12 two-hour time blocks. The purpose of this research is to develop a model which can determine an optimal SAR network configuration for any two hour time block.

Determining an optimal SAR network configuration is a combinatorially explosive task. For example, if 10 additional receiving stations are to be chosen from 25 candidate stations, there are more than 3 million possible network configurations. If eight HFDF receivers can be assigned to a receiving station and there are 30 candidate frequencies, this results in more than 5 million possible frequency assignments for the HFDF receivers at any station. If there are 30 frequencies and 10 receiving stations each with eight HFDF receiving assets, there are more than 10^{66} possible SAR network configurations. It is clearly impractical to consider complete enumeration and comparison of all possible network configuration.

The comprehensive multi-time period GSARP has the additional dimension of time which makes it a more complex problem than the single time period GSARP, which is the focus of this research. The multi-time period GSARP involves determining a good feasible

network configuration of receiving stations and HFDF receivers that is robust over all time periods. Eventually a robust single time period solution methodology which produces good configurations can be expanded to consider the multi-time period GSARP.

3.2 Notation

The following notation is used for the nonlinear and linear model formulations of the GSARP.

Subscripts: Let i index transmitting locations
j, h index receiving station locations
k index frequency bands

Decision Variables:

$$X_j = \begin{cases} 1 & \text{if a receiving station is located at } j \\ 0 & \text{otherwise} \end{cases}$$

$$X_{jk} = \begin{cases} 1 & \text{if an HFDF receiver is located at station } j \text{ assigned to frequency } k \\ 0 & \text{otherwise} \end{cases}$$

Z_j = the number of bundles of HFDF receivers allocated to station j. Each bundle has eight HFDF receivers.

Y_k = the number of HFDF receivers placed on frequency k which exceed the fairshare of resources for that frequency.

Probabilities:

$U_{i\alpha k}(X)$ = the probability of a signal from location i propagating to a particular combination of stations, α.

F_{ik} = the probability that a distress signal emanating from distress location i is broadcasting on frequency k.

P_{ijk} = the probability that a distress signal emanating from location i propagates to station j on frequency k.

W_{ij} = the probability that a line of bearing from station j is within the acceptable circularized error region defined for transmitting location i.

Other Notation:

F = the set of stations belonging to the fixed base network of receiving stations already located in the SAR network.

d_i = the acceptable circularized error radius defined for transmitter i.

C = the set of all combinations of three or more receiving stations.

α = any combination of three or more receiving stations.

$$I_{\alpha i} = \begin{cases} 1 & \text{if combination } \alpha \text{ yields a confidence region radius less than or equal to the acceptable error radius of transmitter location } i. \\ 0 & \text{otherwise} \end{cases}$$

FS = the fairshare of HFDF receivers for each frequency

NB = the total number of bundles of HFDF receivers

NS = the total number of receiving stations

3.3 Nonlinear Formulation

This section presents the nonlinear mathematical formulation of the GSARP adapted from Drake's formulation for the frequency assignment problem (11). Drake's nonlinear formulation can be stated as: Maximize the number of expected geolocations for the SAR network which is subject to a fixed base network of HFDF receiving stations, limited additional HFDF receiving stations, limited HFDF receivers, allocation of HFDF receivers in multiples of eight, allocation of HFDF receivers only to locations which have an HFDF receiving station, and all variables binary or integer. Three lines of bearing and a confidence region radius of less than d_i are required for potential geolocation of transmitter i.

3.3.1 Nonlinear Objective Function. The nonlinear objective function is the probability of a distress signal transmission, F_{ik}, summed over all transmitter locations, i, and frequencies, k, multiplied by the probability of a signal propagating to a particular combination of stations, $U_{i\alpha k}(X)$, summed over all combinations of three or more stations, C, multiplied by an indicator variable, $I_{i\alpha}$, that indicates whether a combination of stations produces a circularized confidence region with a radius less than d_i. The probability, $U_{i\alpha k}(X)$, of a signal propagating to a particular combination of stations is the probability that the signal propagates to all stations in the combination multiplied by the the HFDF binary decision variables for that combination, all multiplied by the probability that the signal does not propagate to any other stations multiplied by the HFDF binary decision variables for these other stations not in the combination. In mathematical terms,

$$U_{i\alpha k}(X) = \left[\prod_{j\in\alpha} P_{ijk}X_{jk}\right]\left[\prod_{h\notin\alpha}(1 - P_{ihk}X_{hk})\right].$$

Using $U_{i\alpha k}(X)$, the complete nonlinear objective function is defined as:

$$\max \quad \sum_{i}^{I}\sum_{k}^{K} F_{ik}\sum_{\alpha\in C} U_{i\alpha k}(X)I_{i\alpha}.$$

3.3.2 Constraints. The nonlinear formulation has six constraints. The first constraint ensures the fixed base network of HFDF receiving stations is not disturbed:

$$X_j = 1, \quad \forall\, j \in F.$$

The second constraint ensures that the number of HFDF receiving stations located does not exceed the total number of HFDF receiving stations available:

$$\sum_{j}^{J} X_j \leq NS.$$

The allocation of HFDF receivers in multiples (bundles) of eight is handled by the third constraint:

$$\sum_{k}^{K} X_{jk} - 8Z_j = 0, \quad \forall\, j.$$

The fourth constraint was developed so that the number of bundles of HFDF receivers located does not exceed the total number of HFDF receivers available:

$$\sum_{j}^{J} Z_j \leq NB.$$

The fifth constraint prevents the allocation of an HFDF receiver to a location which does not have a station:

$$X_{jk} - X_j \leq 0, \quad \forall\, j, k.$$

All variables are restricted to be integer or strictly binary by constraint six:

$$X_j \in \{0, 1\}$$

$$X_{jk} \in \{0, 1\}$$

$$Z_j = \text{integer}.$$

3.3.3 Understanding the Nonlinear Formulation. A careful look at the nonlinear formulation shows that expected behavior of a SAR network is modeled. In a SAR network there is declining utility for each additional HFDF receiver on a frequency that already has adequate coverage (6). When coverage for a frequency is maximized, there is no utility in assigning additional HFDF receivers on that frequency. A study of the nonlinear objective function shows the following items to be true.

- Less than primary coverage for a frequency results in no contribution to the objective function.
- Primary coverage consisting of three HFDF receivers assigned to a frequency only contributes positively to objective function.
- Further coverage consisting of primary coverage plus additional HFDF receiver(s) assigned to a frequency contributes positively to the objective function.
- Frequency saturation points for frequencies may exist where putting more resources on that frequency results in almost no additional expected geolocations for the network.
- During the allocation of an HFDF resource the nonlinear objective considers all possible allocations. The difference in the sum total of probabilities with and without an HFDF resource decision variable X_{jk} represents the number of additional expected geolocations resulting from the allocation of an HFDF to X_{jk}.

3.3.4 Computational Feasibility of the Nonlinear Formulation. The nonlinear formulation is a robust description of the GSARP which seeks to maximize expected geolocations. This formulation is robust, because it explicitly addresses the combinatorial nature of the GSARP. Unfortunately, the nonlinear formulation has an explosive number of nonlinear terms for even small SAR networks. For example, a location problem with just ten receiving station candidates and five frequencies results in 5060 nonlinear terms in the objective function. Binary decision variables also complicate the problem, because some type of implicit enumeration scheme must be used to find an optimal solution. A rule of thumb bounding the number of possible solutions for problems with binary variables

requiring implicit enumeration is 2^n where n is the number of integer variables. The number of possible solutions to be enumerated becomes unreasonably large very quickly as the number of variables, n, increases. Clearly this type of formulation is impractical even for small problems. A simplified formulation is required which embodies characteristics of the inherent nonlinearities and seeks to directly or indirectly maximize expected geolocations.

3.4 Multiobjective Linear Integer Formulation

For computational tractability, the GSARP is formulated as a multiobjective linear integer programming (MOLIP) model. The following paragraphs present a summary of the literature motivating the formulation and descriptions of the accuracy weighting function, objective function, and constraints.

3.4.1 Summary of Relevant Literature. There are three properties of the nonlinear objective function which characterize the SAR network behavior.

- The objective function measures the number of additional expected geolocations resulting from each potential location assignment.
- The objective function shows the declining utility (or decrease in additional expected geolocations) for each additional HFDF receiver on a frequency that already has adequate coverage.
- The objective function recognizes frequency saturation points where putting more resources on that frequency results in almost no additional expected geolocations for the network.

From the literature review there were three ideas which influence the GSARP's formulation and have potential to model some of the characteristic SAR network behavior.

The first concept is Hogan and Revelle's maximum-coverage model BACOP2 (13). The model has two desirable characteristics which can be implemented in a simplified formulation for the GSARP. First, the multiobjective nature of the model allows the optimum coverage configuration to represent different levels of coverage being traded off against one

another. Second, the weighting of the variables rewards coverage levels to higher demands more than that to lower demands.

The second promising concept is Johnson's multiobjective formulation for the HFDF frequency assignment problem which is closely related to the GSARP and, consequently, provides significant insight for model formulation (14). Johnson's formulation has three objectives which are outlined in the next few paragraphs. The first and the third objective functions can easily be incorporated directly or indirectly into the GSARP formulation. On the other hand, the second objective appears to be highly correlated with the first objective while its importance is questionable.

Johnson's first objective essentially maximizes the expected number of lines of bearings for the network, assuming a line of bearing is taken for every signal that transmits and propagates. In mathematical terms,

$$\max \sum_{i}^{I}\sum_{j}^{J}\sum_{k}^{K} F_{ik}P_{ijk}X_{jk}$$

Three or more lines of bearing are required for geolocation of a specific signal. Johnson asserts that maximizing the number of lines of bearing provided for the various transmissions in a fixed network of receiving stations should tend to maximize the number of signals that can be geolocated (14:16). This objective function can be thought of as an indirect way of maximizing the number of geolocations for the SAR network.

Johnson's second objective function which maximizes transmissions was also considered for the HFDF location problem (14). However, there are two points that support not using this objective in the GSARP formulation.

The first point for not using Johnson's second objective is that it is implicitly maximized within the first objective function which maximizes the expected number of lines of bearings for the network. In a sense, the second objective can be considered less important than the first objective. In a recent study, Olson and Dillinger found the omission of less important objectives had little impact on the results of multiobjective optimization problems (20:9-10).

A second point against the use of Johnson's second objective is that it is highly correlated with the first objective function due to the objectives having several variables in common. Correlated objective functions are undesirable, because the weighting between objective functions does not always behave intuitively. Sometimes a very important objective will have a very small weight for the best solution. The angle between the cost vectors is an acceptable way to measure the correlation between objective functions (23:198). The smaller the angle, the larger the correlation. The correlation between objectives one and two was confirmed during a case study where the angle of correlation between Johnson's first and second objective functions was measured to be 4.3296 degrees (15:11). In mathematical terms, Johnson's second objective is:

$$\max \quad \sum_{i}^{I}\sum_{j}^{J}\sum_{k}^{K} F_{ik}X_{jk}.$$

Johnson's third objective function penalizes excessive coverage by HFDF receivers of a frequency (14:17-19). This objective recognizes the declining utility (decrease in additional expected geolocations) for each additional HFDF receiver on a frequency that already has adequate coverage. It can be thought of as an equity maximizing objective, since the objective will not penalize equal coverage. By evaluating the objective functions several times and giving varying levels of importance to the excess coverage objective, there is potential for revealing the optimal tradeoff between the objectives which will maximize geolocations. Searching for the optimum tradeoff of frequency coverage using multiple objectives is effectively a heuristic way for searching the nonlinear decision space. In mathematical terms, Johnson's third objective is:

$$\min \sum_{k}^{K} Y_k.$$

The third promising concept is the declining distant-dependent function discussed by Pirkul and Schilling (21:140-153). Their model attempts to provide acceptable assignments for uncovered demands by using a declining distance-dependent function for demand assignments that exceed the acceptable coverage distance. For the GSARP, a similar weighting

function will be developed to recognize the negative effect of assigning an HFDF resource to a facility that has large lines of bearing (LOB) errors. This accuracy-dependent weighting function is described in the next section.

3.4.2 Declining Accuracy-Weighting Function. An accuracy-weighting function was developed from a technical paper which documents confidence area mathematics and assumptions associated with position finding (12). Consider a LOB from receiving station j to transmitter i. Several assumptions concerning HFDF confidence area mathematics are reasonable to make for the HFDF location problem (6):

1. Projections of the LOBs are straight lines.
2. The earth is flat near target position T_i.
3. The angular bearing error of the LOB taken from receiving station j to transmitter i is σ_{ij} is normally distributed with a mean of zero and a variance of σ_{ij}^2.
4. The width of the bearing fan, e_{ij}, is $(R_{ij} \times \sin \sigma_{ij})$ which is normally distributed with a mean of zero and a variance of e_{ij}^2, since σ_{ij} is normally distributed. R_{ij} is the range from station j to transmitter i.

It is also true for the GSARP that an acceptable circularized error radius d_i is known for transmitter i. From this information the following declining accuracy-dependent function can be developed:

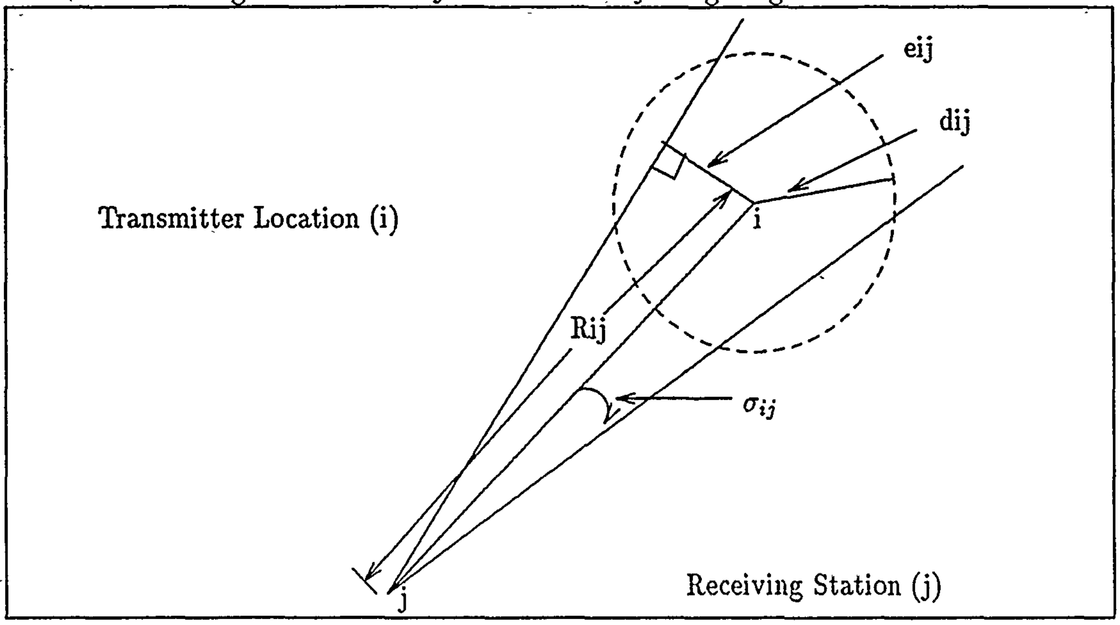
$$z_{ij} = \frac{d_i}{e_{ij}}$$

where z_{ij} is a standard normal random variate which equals the number of standard deviations of bearing fan width that are within the acceptable circularized error radius d_i. The resulting weighting function, W_{ij}, equals the probability that the bearing fan e_{ij} is less than d_i:

$$W_{ij} = 1 - 2[1 - \Phi(z_{ij})].$$

where $\Phi(z_{ij})$ is a function of the percentile of the standard normal random variate. The geometry of this weighting function from receiving station j to transmitter location i can be seen in Figure 1.

Figure 1. Geometry of the Accuracy Weighting Function



3.4.3 Multiobjective Linear Objective Functions. The GSARP has two opposing objectives when it is formulated as a multiobjective linear integer program (MOLIP). These objectives were adapted from Johnson's formulation of the frequency assignment problem (14). The first objective maximizes the number of expected lines of bearings while the second objective minimizes the excess coverage of frequencies.

3.4.3.1 Objective Function One. The first objective function is designed to maximize the expected number of SAR network geolocations by maximizing the expected number of accurate LOBs for the SAR network. A LOB is generated when a signal transmits and propagates from a transmitter location to a receiving station on a given frequency. The probability of generating a LOB from station j to transmitter i on frequency k is multiplied by the probability of generating an accurate LOB from station j to transmitter location i, W_{ij}. This objective is similar to Johnson's first objective for the HFDF frequency assignment problem with the accuracy of the LOBs incorporated (14:16-17).

$$\max \sum_{i}^{I} \sum_{j}^{J} \sum_{k}^{K} W_{ij} F_{ik} P_{ijk} X_{jk}$$

3.4.3.2 Objective Function Two. The second objective function is identical to Johnson's excess coverage objective which penalizes excessive coverage of a frequency by HFDF receivers (14:17-19). This objective recognizes the declining utility (decrease in additional expected geolocations) for each additional HFDF receiver on a frequency that already has adequate coverage. Excess coverage is defined as anything more than the fair share. Fair share is defined as the total number of HFDF receivers divided by the total number of frequencies rounded to the next largest integer. In mathematical terms, the second objective is,

$$\min \sum_{k}^{K} Y_k.$$

3.4.4 Constraints. The MOLIP formulation has eight constraints. The first constraint ensures the fixed base network of HFDF receiving stations is not disturbed.

$$X_j = 1, \quad \forall\, j \in F$$

The second constraint ensures that the number of HFDF receiving stations located does not exceed the total number of HFDF receiving stations available:

$$\sum_{j}^{J} X_j \leq NS$$

HFDF receivers are allocated in multiples (bundles) of eight by the third constraint:

$$\sum_{k}^{K} X_{jk} - 8Z_j = 0, \quad \forall\, j.$$

The fourth constraint prevents the number of bundles of HFDF receivers from exceeding the total number of HFDF receivers available:

$$\sum_{j}^{J} Z_j \leq NB.$$

The fifth constraint ensures that an HFDF receiver is not allocated to a location which does not have a station:

$$X_{jk} - X_j \leq 0, \quad \forall\, j,k.$$

Constraint six is a *quasi-covering* constraint which guarantees either no coverage at all, or at least primary coverage . Johnson's formulation of the HFDF assignment problem included a full covering constraint. However, it may not be logical to explicitly require primary coverage for all frequencies since the probability of transmission may be close to zero for some frequencies. Since a minimum of three signals are required for a geolocation, the quasi covering constraint prevents the wasteful allocation of just one or two resources to a frequency. This mimics the nonlinear objective function which receives no contribution for less than primary coverage:

$$\sum_{j \in s} X_{jk} - 2X_{hk} \leq; 0, \quad \forall\, h \notin s, j, k.$$

$$\text{where} : s \in S : \left\{ \text{all combinations} \begin{pmatrix} J \\ (J-1) \end{pmatrix} \text{and } J = NS \right\}.$$

With the seventh constraint, the structural excess coverage variable is assigned the value of excess coverage allocated to frequency k:

$$\sum_{j} X_{jk} - Y_k \leq FS \quad \forall\, k.$$

The eighth constraint restricts all variables to be binary integer or integer:

$$\begin{aligned} X_j &\in \{0,1\} \\ X_{jk} &\in \{0,1\} \\ Z_j &\in \{0,1,2,3\} \\ Y_k &\geq 0 \text{ and integer.} \end{aligned}$$

3.4.5 The Complete Multiobjective Linear Formulation. The complete problem to be analyzed and solved is:

$$\max \quad \sum_{i}^{I}\sum_{j}^{J}\sum_{k}^{K} W_{ij}F_{ik}P_{ijk}X_{jk} \tag{1}$$

$$\min \quad \sum_{k}^{K} Y_k \tag{2}$$

subject to

$$\begin{aligned}
X_j &= 1, \quad \forall\, j \in F && (3)\\
\sum_{j}^{J} X_j &\leq NS && (4)\\
\sum_{k}^{K} X_{jk} - 8Z_j &= 0, \quad \forall\, j && (5)\\
\sum_{j}^{J} Z_j &\leq NB && (6)\\
X_{jk} - X_j &\leq 0, \quad \forall\, j,k && (7)\\
\sum_{j \in s} X_{jk} - 2X_{hk} &\leq 0, \quad \forall\, j,k,h \notin s && (8)\\
\sum_{j}^{J} X_{jk} - Y_k &\leq FS, \quad \forall\, k && (9)\\
X_j &\in \{0,1\} && (10)\\
X_{jk} &\in \{0,1\} && (11)\\
Z_j &\in \{0,1,2,3\} && (12)\\
Y_k &\geq 0 \text{ and integer.} && (13)
\end{aligned}$$

3.4.6 Computational Tractability. The multiobjective linear integer formulation has significantly fewer terms and variables than the nonlinear formulation. For J stations and K frequencies there are $J \times K + J + K$ integer variables of which $J \times K + J$ variables are binary. For example, a problem with 10 receiving station candidates and five frequencies results in just 50 linear objective function terms compared to 5060 terms for

the nonlinear formulation.

The next chapter, Computational Experience with Model Formulations, will investigate the nonlinear and multiobjective linear integer formulations in more detail by comparing test case problem results for the two formulations.

IV. Computational Experience with Model Formulations

4.1 Overview of Test Cases

Two small-scale test problems were studied to document any relationship found between the nonlinear integer program (NLIP) and the multiobjective linear integer program (MOLIP) formulations presented in Chapter 3. Size limitations, data selection, and result comparisons are discussed for test cases in the next three subsections.

4.1.1 Size Limitations. The size of the test cases was limited for two reasons. The first reason is that the computational complexity of the nonlinear objective function coefficients becomes combinatorially explosive as the problem size increases. Secondly, problem size is limited, because the nonlinear integer optimization code's enumeration scheme makes large problems computationally undesirable.

Both test cases used five receiving station locations and four transmitter locations. The first test case used three frequencies while the second test case used five frequencies. This resulted in the first test case having 20 and 23 variables for the NLIP and MOLIP formulations respectively. The second test case had 30 and 35 variables for the NLIP and MOLIP respectively. The number of nonlinear objective function terms was 48 for the first test case and 80 for the second test case. Using just five receiving station locations kept the number of variables, the number of terms, and the computational complexity to a minimum.

4.1.2 Data Selection. The data used in both test case problems is a subset of the data provided for computations by the Department of Defense. While the NLIP formulation explicitly considers accuracy for a combination of stations to a transmitter, the MOLIP formulation considers only the accuracy for a single line of bearing from a receiving station to a transmitter. For this reason, test case problem data was selected with one or more combinations of receiving stations having an unacceptable confidence region accuracy for transmitter(s) in the NLIP formulation. The performance of the MOLIP is studied with this data, to see if the MOLIP can find solutions comparable to the NLIP

even though it does not have any information about the unacceptable accuracy of one or more confidence regions.

4.1.3 Comparison of Results. The solutions of both the NLIP and MOLIP formulations are compared by evaluating both solutions using the nonlinear objective function. The relative percent difference in results is computed by using the formula below.

$$\frac{(\text{MOLIPsolution} - \text{NLIPsolution})}{\text{NLIPsolution}} \times 100\%$$

4.2 Nonlinear Integer Programming (NLIP)

This section presents the modified NLIP formulation and the NLIP algorithm used to solve the test case problems.

4.2.1 NLIP Solution Algorithm. The NLIP solution algorithm uses an implicit enumeration scheme designed to specifically handle nonlinear programming problems with linear constraints (4). The NLIP algorithm generates all feasible incumbent solutions that are encountered in the course of finding the solution. A feasible incumbent solution is an intermediate solution that is the best feasible solution during a particular stage of the optimization search. At any time during the search, the current feasible incumbent solution represents a lower bound for the optimal solution. During the optimization procedure, solutions are rejected when they are less than the current incumbent solution or upper bound. Monotonicity of the objective function is required for this search to always find the optimal solution. Monotonicity has not been shown for the NLIP; however, for the test problems, the NLIP algorithm converged to the same solution for every starting point that was used.

A matrix generator is used to create the input file required by the nonlinear zero-one solver. A copy of the optimization code, input files, matrix generator code, and toy problem data files can be found on the floppy disk one that accompanies this thesis. The files for the two toy problems can be found in two directories named *toy1* and *toy2* on floppy disk 1.

4.2.2 NLIP Test Case Formulation. The NLIP formulation discussed in detail in Chapter 3 is summarized again with respect to the specific NLIP formulation used for the test cases. Due to size limitations, HFDFs are dealt with as single entities rather than as bundles of entities. This modification results in constraints three and four being different from what was presented in section 3.3.

$$\max \quad \sum_{i}^{I}\sum_{k}^{K} F_{ik} \sum_{\alpha \in C} U_{i\alpha k}(X) I_{i\alpha}$$

$$\text{where} \quad U_{i\alpha k}(X) = \left[\prod_{j \in \alpha} P_{ijk} X_{jk}\right]\left[\prod_{h \notin \alpha}(1 - P_{ihk} X_{hk})\right]$$

subject to

$$\begin{aligned} X_j &= 1. \quad \forall\, j \in F \\ \sum_{j}^{J} X_j &\leq NS \\ \sum_{j}^{J}\sum_{k}^{K} X_{jk} &\leq NH \end{aligned}$$

where NH is the number of HFDF receivers

$$\begin{aligned} X_{jk} - X_j &\leq 0, \quad \forall\, j, k \\ X_j &\in \{0, 1\} \\ X_{jk} &\in \{0, 1\}. \end{aligned}$$

4.3 Multiobjective Linear Integer Programming (MOLIP)

This section presents the modified MOLIP formulation and the MOLIP algorithm used to solve the test case problems.

4.3.1 MOLIP Solution Algorithm The ADBASE multiple objective linear programming package was used to identify the efficient frontier by computing all *efficient extreme points* associated with the solution space of each test case (22). An efficient extreme point is also referred to as a *pareto-optimal* solution. A pareto-optimal solution is a feasible solution which is as good as or better than all other viable solutions for the multiple objective problem. It is the pareto optimal solutions generated by ADBASE which are compared to the NLIP results. For each extreme point, ADBASE can also provide the relative cost for each variable not in a current solution. From these costs the relative weighting or importance of each objective function can be determined for an extreme point.

A copy of the input files, coefficient generator codes and test case problem data files can be found on the floppy disk which accompanies this thesis. The files for the two toy problems can be found in the two directories named *toy*1 and *toy*2 on floppy disk 1.

4.3.2 MOLIP Test Case Formulation. The MOLIP formulation discussed in detail in Chapter 3 is summarized again with respect to the specific MOLIP formulation used for the test case. Due to test case problem size limitations, HFDFs are dealt with as single entities rather than as bundles of entities. This modification results in constraints three and four being different from what was presented in section 3.4. Another limitation in solving the test case is caused by the fact that ADBASE is not a mixed integer programming package. The quasi-covering constraint which is equation 8 presented in section 3.4.5, required that either zero or greater than three HFDFs be assigned to a frequency; however, this constraint prevents ADBASE from finding an integer solution. As a result, the quasi-covering constraint is not included so that ADBASE can be used to identify the entire efficient frontier for comparison with the NLIP solution.

In order to clearly identify the relative weightings of the objective, the two objective functions had to be normalized. Multiplying the first objective function by 100 makes both objective functions closer to the same order of magnitude. ADBASE is optimizing the normalized problem when it computes all efficient extreme points.

$$\max \sum_{i}^{I}\sum_{j}^{J}\sum_{k}^{K} W_{ij}F_{ik}P_{ijk}X_{jk}$$

$$\min \sum_{k}^{K} Y_k$$

subject to

$$\begin{aligned}
X_j &= 1, \quad \forall\, j \in F \\
\sum_{j}^{J} X_j &\leq NS \\
\sum_{j}^{J}\sum_{k}^{K} X_{jk} &\leq NH
\end{aligned}$$

where NH is the number of HFDF receivers

$$\begin{aligned}
X_{jk} - X_j &\leq 0, \quad \forall\, j,\, k \\
\sum_{j}^{J} X_{jk} - Y_k &\leq FS, \quad \forall\, k \\
X_j &\in \{0,1\} \\
X_{jk} &\in \{0,1\} \\
Y_k &\geq 0 \text{ and integer.}
\end{aligned}$$

4.4 Test Case 1 Results

For test case 1, the NLIP algorithm identified 13 feasible incumbent solutions before the algorithm found the optimal solution of 0.1137217. ADBASE found four efficient extreme points. However three of these extreme points can effectively be collapsed to provide the same solution. The only difference in these three extreme points is that one of them has ST3 with no HFDF resources assigned, one of them has ST4 with no resources assigned, and the last one doesn't have ST3 or ST4. These three solutions are in effect the same, because locating a receiving station without any resources assigned contributes

nothing to the objective function. As a result, the four efficient extreme points are collapsed to provide two unique pareto-optimal solutions which correspond to two different levels of coverage as defined by the second objective function. These two pareto-optimal solutions are evaluated using the nonlinear objective function providing pareto-optimal solutions of 0.10821046 and 0.1127499. It is interesting to note that the second solution of 0.1127499 corresponds to the eleventh feasible incumbent solution for the NLIP. The NLIP algorithm's computations took about one CPU minute and the ADBASE algorithm took less than one CPU minute, both on a VAX 11/785. Convergence of the NLIP algorithm was not significantly improved when the best MOLIP solution was used as a starting point. The two pareto-optimal solutions for the MOLIP along with NLIP optimal solution are compared in the Table 1. For test case 1, the specific formulation and data, as well as the formatted input files for the MOLIP and NLIP solvers are all presented in Appendix G.

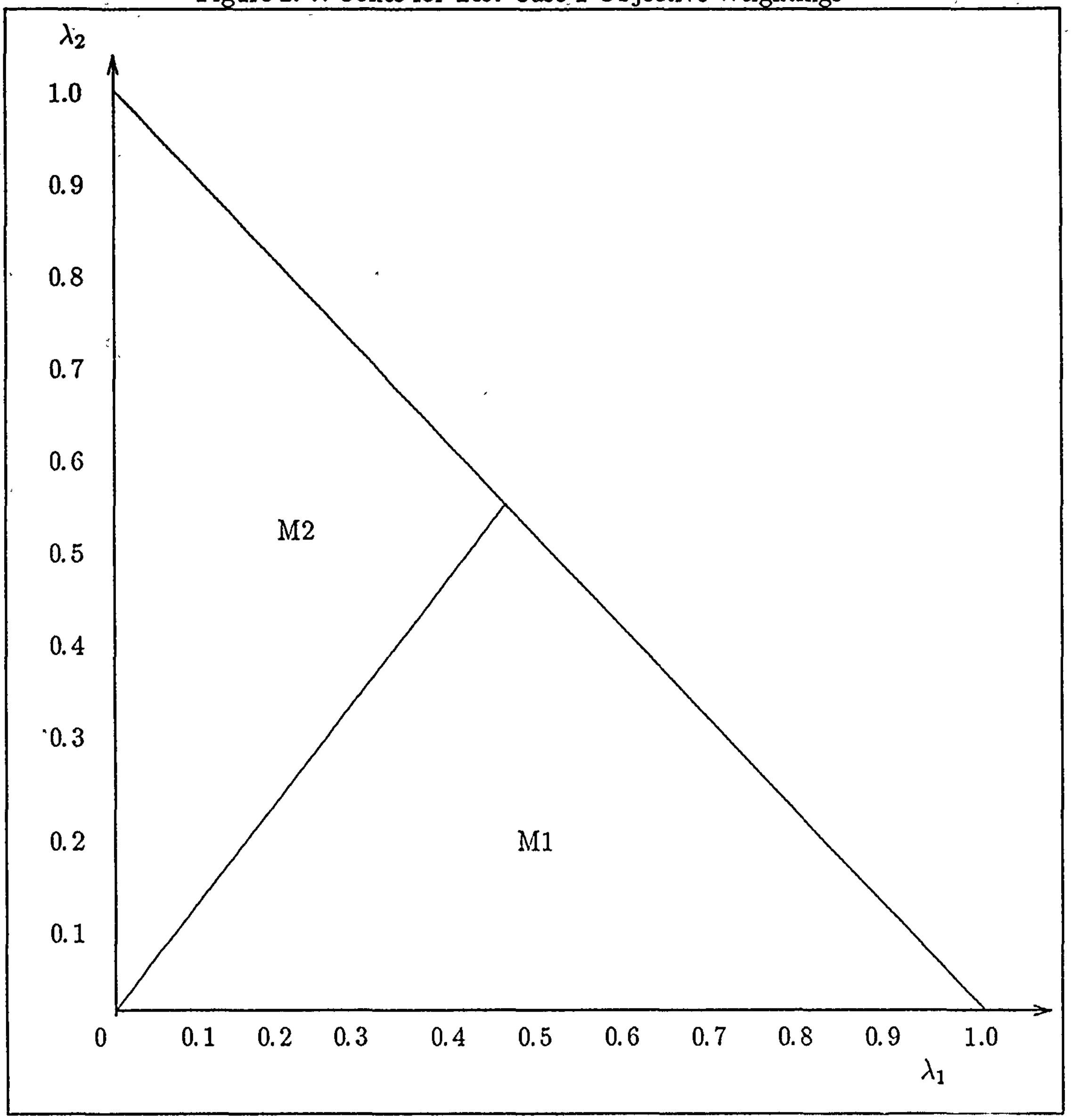
Table 1. Test Case 1 Results

Type Solution	λ_1	Stations	Station HFDFs	NLIP obj	% diff
true NLIP	N/A	ST1	F1 F2 F3	0.1137217	none
		ST2	F1 F2 F3		
		ST4	F2 F3		
		ST5	F1 F3		
M1	$\lambda_1 > 0.439$	ST1	F1 F2 F3	0.1127499	0.85%
		ST2	F1 F2 F3		
		ST3	F2		
		ST5	F1 F2 F3		
M2	$\lambda_1 < 0.439$	ST1	F1 F2 F3	0.1082105	4.86%
		ST2	F1 F2 F3		
		ST3 or ST4	none		
		ST5	F1 F2 F3		

M2 represents the pareto optimal solution where three efficient extreme points collapse to the same solution. The impact of the second objective function can be seen in M2. This solution does not allow excess coverage for any of the frequencies which, for the first test case, is defined as coverage beyond three HFDFs for a frequency.

Figure 2 is a graph depicting the relative impact of objectives one and two for the two MOLIP solutions. The line connecting λ_1 and λ_2 represents the line of objective weightings

Figure 2. λ-Cones for Test Case 1 Objective Weightings



between objective one and objective two where λ_1 is the weighting for objective function one and λ_2 is the weighting for objective function two. It is clear from the graph that λ_1 added to λ_2 must equal one. All of the weightings where λ_1 is less than 0.4388 correspond to M2 and all the weightings where λ_1 is greater than 0.4388 correspond to M1.

The small size of this test case makes it difficult to draw generalizations about how successful the MOLIP formulation will be in providing a good solution for the GSARP. However, three observations can be made. First, the MOLIP solution is relatively close to the NLIP solution. Second, a higher weighting on objective one is preferred for the best solution. Third, the MOLIP solution will provide two or more pareto optimal GSARP network solutions which can be selected based on their relative NLIP solutions.

4.5 Test Case 2 Results

Test case 2 represents a slightly larger problem. The difference is that test case two allocates HFDFs to five frequencies rather than three. For this test case, the NLIP algorithm identified 28 feasible incumbent solutions before the algorithm found the optimal solution of 0.1440151. ADBASE found three efficient extreme points which correspond to different levels of coverage determined by the second objective function. The pareto-optimal solutions are evaluated using the nonlinear objective function. The corresponding MOLIP optimal solutions are 0.12465, 0.10547, and 0.093934

For this problem, the ADBASE solution was not a feasible incumbent solution for the NLIP. However, the best MOLIP solution lies between the objective value of the 19th and 20th incumbent solutions of the NLIP. The increased size of this test case significantly affected the CPU time required for the NLIP solution algorithm. The NLIP optimization code took 4 hours and 55 minutes of CPU time on a VAX 11/785, whereas ADBASE code still took just one or two minutes of CPU time. When the best MOLIP solution was used as a starting solution for the NLIP, the solution time was decreased by only 29 minutes or 10%.

The impact of the second objective function can be seen in M2 and M3, since the level of excess coverage is less than in M1. For example, M3 foı test case two does not

allow excess coverage for any of the frequencies.

Table 2. Test Case 2 Results

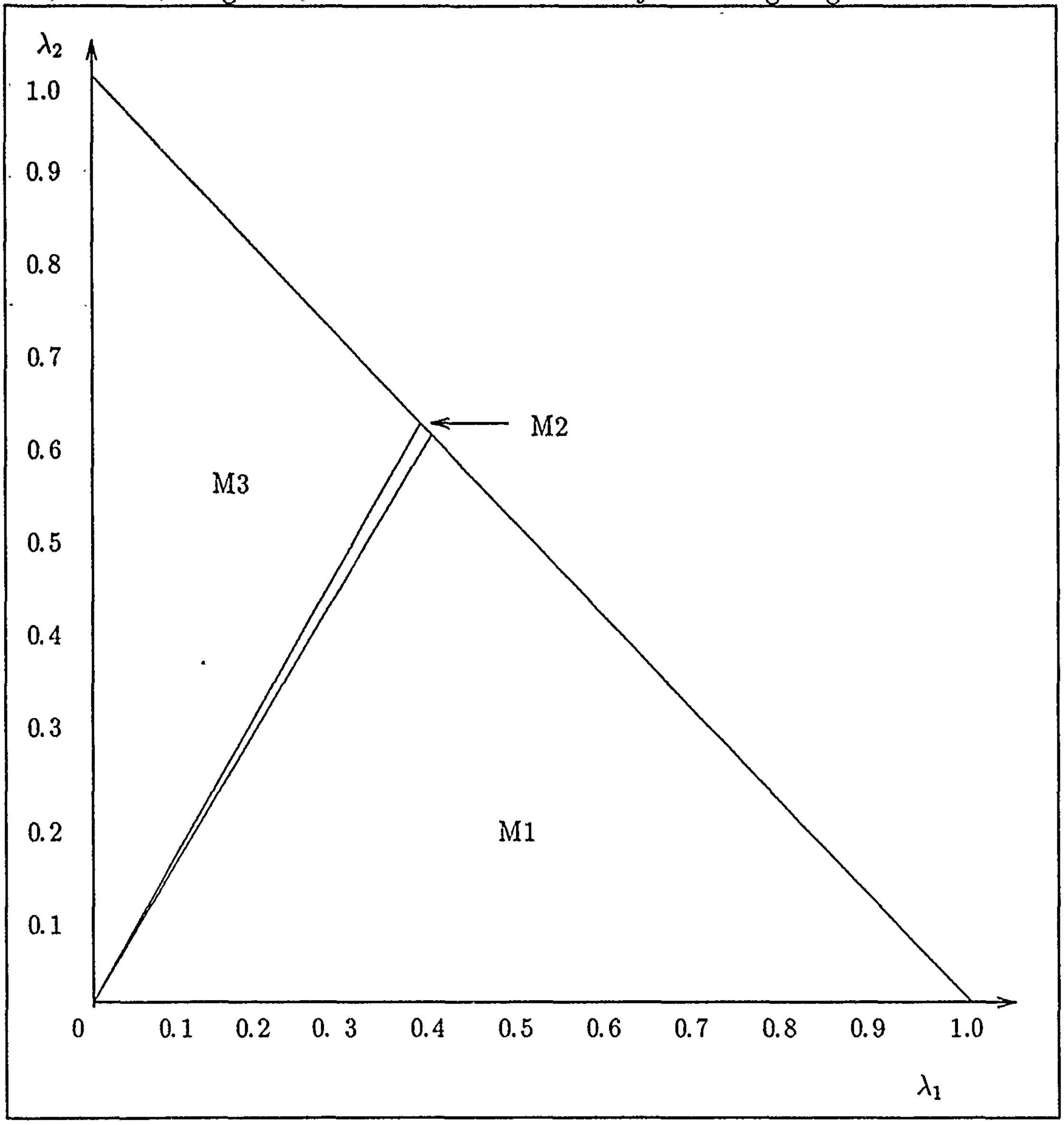
Type Solution	λ_1	Stations	Station HFDFs	NLIP obj	% diff
true NLIP	N/A	ST1	F2 F3 F4 F5	0.1440151	none
		ST2	F2 F3 F4 F5		
		ST3	F2 F3 F4 F5		
		ST5	F3 F4 F5		
M1	$\lambda_1 > 0.383$	ST1	F1 F2 F3 F4 F5	0.1246539	13.444%
		ST2	F3 F4 F5		
		ST4	F1 F2 F3 F4 F5		
		ST5	F4 F5		
M2	$\lambda_1 < 0.383$	ST1	F1 F2 F3 F4	0.1054576	26.773%
	and	ST2	F2 F3 F4 F5		
	$\lambda_1 > 0.362$	ST4	F1 F2 F3 F4 F5		
		ST5	F1 F4		
M3	$\lambda_1 < 0.362$	ST1	F1 F2 F3	0.0939340	34.774%
		ST2	F2 F3 F4 F5		
		ST4	F1 F2 F3 F4 F5		
		ST5	F1 F4 F5		

The configuration of the GSARP network changes quite a bit for each pareto-optimal solution. The fact that the configuration is changing as the excess coverage objective weighting increases makes it possible for the MOLIP to expose several possible configurations representing different levels of coverage. As a result, several pareto-optimal configurations can be evaluated as possible solutions to the true GSARP.

In Figure 3 is a graph depicting the relative impact of objective one and objective two for the two MOLIP solutions. The line connecting λ_1 and λ_2 represents the line of objective weightings between objective one and objective two where λ_1 is the weighting for objective function one and λ_2 is the weighting for objective function two. Once again, it is clear from the graph that λ_1 added to λ_2 must equal one. Notice all of the weightings where λ_1 is less than 0.36237 correspond to M3 and all the weightings where λ_1 is greater than 0.36237 and less than 0.382776 correspond to M2. There is no clear explanation for why M2 commands such a small λ-cone. M2 and M3 both represent pareto-optimal solutions where objective two is more heavily weighted. The impact of objective two, which penalized excess coverage, is reflected in the solutions that are presented in Table 2.

Finally, all the weightings where λ_1 is greater than 0.38278 correspond to the best MOLIP solution M1.

Figure 3. λ-Cones for Test Case 2 Objective Weightings



4.6 Observations and Conclusions

Several observations were made from these test cases. First, it is important to remember that the test case problems were limited in size. With this qualification, it can be

observed that the MOLIP results are relatively close to the true NLIP solution for both test cases. Test case one was 0.85% from it's true NLIP, and test case two was 13.44% from it's true NLIP solution. Furthermore, the best solution for both test cases was the one which most heavily weighted the first objective, which maximizes the expected number of accurate lines of bearing. The conclusion that can be drawn from these results is that there is no reason to doubt that the MOLIP might provide good feasible solutions for the larger GSARP.

A second observation is that the practicality of the NLIP solution algorithm decreases with increased problem size. The NLIP for test case two had ten more structural variables than test case one. The addition of ten variables increased the CPU time on a VAX 11/785 from approximately one minute for the first test case, which has 20 structural variables, to nearly five CPU hours for the second test case, which has 30 structural variables. This indicates that the required CPU time is unmanageable for the larger GSARP which has 991 structural variables as defined by the formulation in Chapter 3.

Another observation is that the GSARP network configuration solutions change as the weighting on the excess coverage objective is increased. This allows the MOLIP to expose several solutions representing different levels of coverage. The result is several pareto-optimal solutions are produced that can be evaluated and compared as potential solutions to the GSARP.

Integrality problems exist with the MOLIP that are not addressed with these test cases. These integrality problems associated with constraints that are not used for the test cases will be addressed in the next chapter, Methodology.

V. Methodology

A multiobjective linear integer programming (MOLIP) formulation was identified as a promising approach for the generalized search and rescue problem (GSARP). This chapter describes the methodologies used to solve the MOLIP formulation of the GSARP. Linear programming, integer programming, multiobjective optimization and network representation are specifically covered.

5.1 Linear Programming

For computational tractability, a linear formulation with both a linear objective function and linear constraints is used to approximate good GSARP solutions in place of the highly nonlinear formulation proposed by Drake (11). A general procedure used for solving linear programming problems is the simplex method. The simplex method involves searching the feasible region boundary defined as the region contained within or on the intersection of the linear constraint equations. The algorithm iteratively moves to better adjacent corner point solutions, which are the points where the constraint equations intersect. When no better adjacent corner point solution can be found, the search is terminated. While the simplex method could theoretically examine every corner point, it has proven (on the average) to be very efficient on most problems of practical origin (18:38). The linear programming software considered for the computations herein utilize variants of the simplex method for solving linear programs.

5.2 Integer Programming

The decision variables for the MOLIP presented in chapter three must be integer. Specifically, the variables X_j and X_{jk} must equal zero or one, while Y_k must be a positive integer greater than or equal to zero.

The subset of constraints used for the test cases in Chapter 4 preserve the integrality of the decision variables, and integer solutions can be found efficiently using linear programming algorithn s. Unfortunately, the remaining subset of constraints destroy the

integrality of the decision variables when they are used with pure linear programming algorithms.

The integer programming technique most often found with commercially available software is the branch-and-bound technique. A generalization of the branch-and-bound algorithm follows (14:22). The branch-and-bound algorithm solves iterative linear programming problems with integer constraints relaxed. At each iteration, the set of feasible solutions is partitioned into subsets (hence branching) and an upper bound is calculated for each subset. When an integer solution is found its value becomes a lower bound for the optimal solution for a maximization problem (hence bounding). A subset is abandoned from further branching if its upper bound is less than the lower bound for the optimal solution. If a subset has either an infeasible solution or an integer solution, it is also abandoned, since no improvement can be found from the subset by further partitioning. Any subset that has been abandoned due to an integer solution, infeasible solution, or an upper bound that is dominated by an integer solution, is said to have been fathomed. Any integer solution found that has a larger upper bound than the incumbent integer lower bound becomes the new incumbent lower bound for the optimal solution. Subsets continue to be partitioned or abandoned until all subsets have been fathomed. At this point, the optimal integer solution is the current incumbent integer lower bound solution, if one exists. To solve problems of practical size, branching and/or bounding rules can be implemented which may improve the performance of the branch-and-bound search (18:252- 258).

The next two subsections discuss the bundling constraints and the quasi-covering constraints which destroy the integrality of the MOLIP solution.

5.2.1 Bundling Constraints. The bundling constraint defined for the GSARP requires that HFDF receivers be allocated in bundles (multiples) of eight to the receiving stations. For the thesis problem, each station can receive either one or two bundles of HFDF receivers. This constraint was considered to be nonnegotiable, since it was requested by the DOD who is sponsoring the research. Therefore, the bundling constraint will be explicitly modeled in the network representations presented in chapter six. The bundling constraint dictates that mixed integer programming be considered in order to

achieve integer solutions.

5.2.2 Quasi-Covering Constraints. The quasi-covering constraint defined for the GSARP requires that a frequency receives at least primary coverage of three HFDF receivers or no coverage at all. This constraint was introduced in an attempt to better mimic the behavior of the nonlinear formulation. This constraint is considered negotiable, since it was not requested by the research sponsor.

From a computational standpoint, this constraint cannot be practically implemented. Using the constraint would require the explicit integerization of all 991 variables introduced in the model formulation. Due to insufficient scratch work space for the branch-and-bound algorithm, experimental runs with just 115 variables using SASLP in the mixed integer mode did not converge to the optimal integer solution. Therefore, alternate means of addressing the quasi-covering are considered.

One way to address the constraint is to allow the linear program to find an optimal solution without using the quasi-covering constraint. This would still prevent unnecessary HFDFs from being assigned to frequencies that have a very low probability of transmission. However, the objective function would still benefit from the assignment of just one or two HFDFs to a frequency, when, in reality, no geolocation would be attempted with less than primary coverage of three HFDFs providing a signal (11:1).

A second way to address the quasi-covering constraint is to use a true covering constraint as Johnson did in her research for the HFDF assignment problem (14). This constraint ensures that the objective function correctly benefits from placement of an HFDF on a frequency, since the constraint requires primary coverage for each frequency. On the other hand, this constraint places at least primary coverage on each frequency even if there is little or no probability of transmission on a frequency.

The bottom line is that both ways of dealing with the intended quasi-covering constraint have the potential to assign HFDFs to frequencies which result in little or no improvement of the true GSARP objective, maximizing the expected number of geolocations in a SAR network.

The strategy for addressing the quasi-covering constraint is brute force. Neither alternative is clearly better than the other; therefore, both alternatives are explored by this research. In other words, thesis results include solutions both with and without the covering constraint in order to document their performance on the large problem.

5.2.3 Integer Programming Software Although several integer branch-and-bound codes were available for use with small problems, only SAS LP was available which could handle problems with greater than 100 variables. SAS LP also has a sparse input structure which is designed to make larger problems easier to input and run more efficiently.

5.3 Multiobjective Optimization

The true objective for the GSARP, maximizing geolocations in a SAR network, is not multiobjective. However, the nonlinear objective proposed by Drake is intractable for large problems. This was confirmed by the larger test case problem presented in chapter four (11). The multiobjective approach described in Chapter 3 can be considered a heuristic for exposing several pareto optimal solutions representing different coverage levels in a SAR network. This research assumes that a multiobjective approach can provide better solutions than either of the single objective functions by themselves.

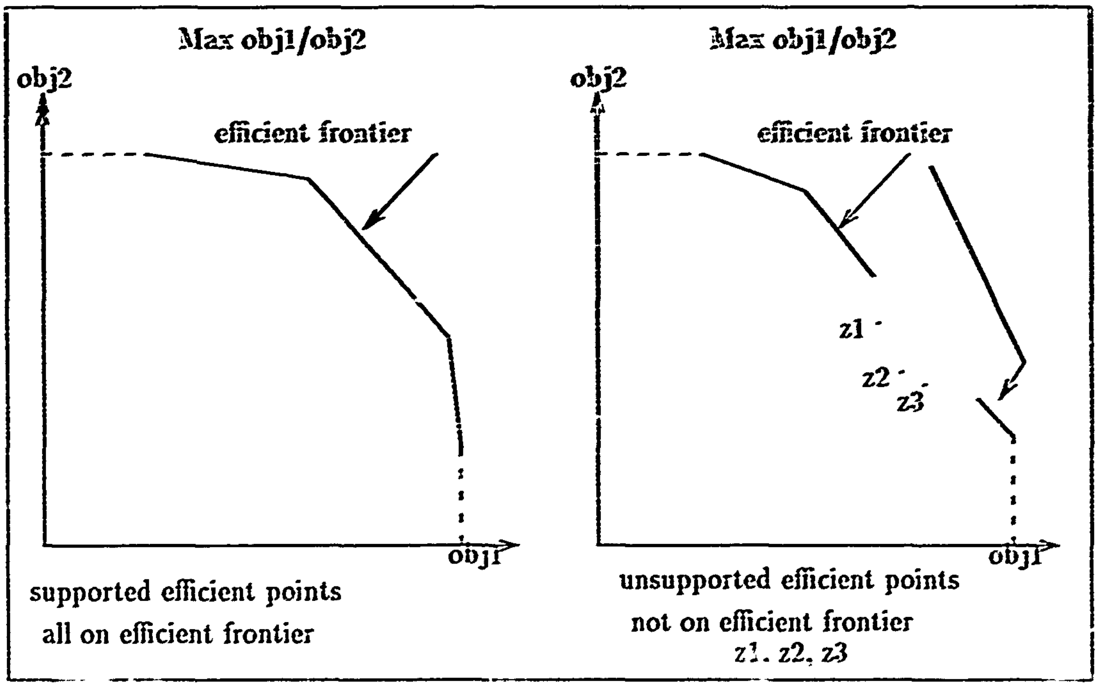
The remainder of this section describes pareto optimality, methods for identifying the efficient frontier, and analysis of the GSARP's efficient frontier.

5.3.1 Pareto-Optimality. A pareto-optimal solution, also referred to as an efficient extreme point, is a feasible solution that is as good as, or better than, all other feasible solutions. From a pareto-optimal point, it is not possible to move in a direction so as to increase one of the objectives without necessarily decreasing another objective. The set of all pareto-optimal points is called the efficient set or the efficient frontier. The efficient frontier must be identified in order to analyze the solutions of a multiobjective optimization problem. Some of the methods used for identifying the efficient fiontier are described in the remaining subsections.

5.3.2 Multiobjective Simplex Method. For linear problems, the multiobjective simplex method can be used to identify the efficient frontier. ADBASE, which uses a multiobjective revised simplex method, is the only software available at AFIT that can identify the entire efficient frontier (22). Although ADBASE was implemented successfully on the test cases presented in chapter four, it cannot be used to solve the thesis problem, since it is not capable of guaranteeing integer solutions to multiobjective linear integer problems.

5.3.3 Supported and Unsupported Pareto-Optimal Points. A pareto-optimal point can be mapped into the criterion space by plotting the values of one objective function versus another. For a multiobjective linear program, the criterion mappings of pareto optimal solutions will form a piece-wise linear convex hull where all of the pareto-optimal points can be found on the border of the convex hull (23:431). Pareto-optimal points found on the border of the convex hull are called *supported* pareto-optimal points (23:431).

Figure 4. Supported versus Unsupported Pareto-Optimal Solutions (23:432)



In multiobjective integer and nonlinear programs, the criterion mapping of a pareto optimal point into the criterion space can lie inside the border of the convex hull (23:432). Pareto-optimal points inside the border of the convex hull are called *unsupported* efficient

points (23:432). Figure 4 shows an example of supported pareto-optimal solutions versus unsupported pareto-optimal solutions. Because unsupported pareto-optimal points are dominated by other convex combinations of the objective functions, they cannot be generated using the weighted sums approach described in the following paragraph.

5.3.4 Weighted Sums Approach. For the weighted sums approach, each objective is multiplied by a strictly positive scalar weight λ_i such that the sum of the λ_is equals one. The weighted objectives are then combined to form a consolidated objective function. The weighted sums approach can be thought of as a method that experiments with strictly convex combinations of the objective functions where it is not obvious what might be an optimal combination of λ_is (23:165). The weighted sums approach has an advantage over a multiobjective simplex method, because it can be implemented with a standard linear programming package which in general, requires less computation. Unfortunately, the set of all possible convex combinations of the objective function is infinite, so there is no guarantee that the entire efficient frontier will be identified using the iterative approach. When the weighted sums approach is used with an integer programming problem, it may not be able to identify the entire efficient frontier since unsupported efficient points which can occur with integer programming cannot be identified with this method (23:433).

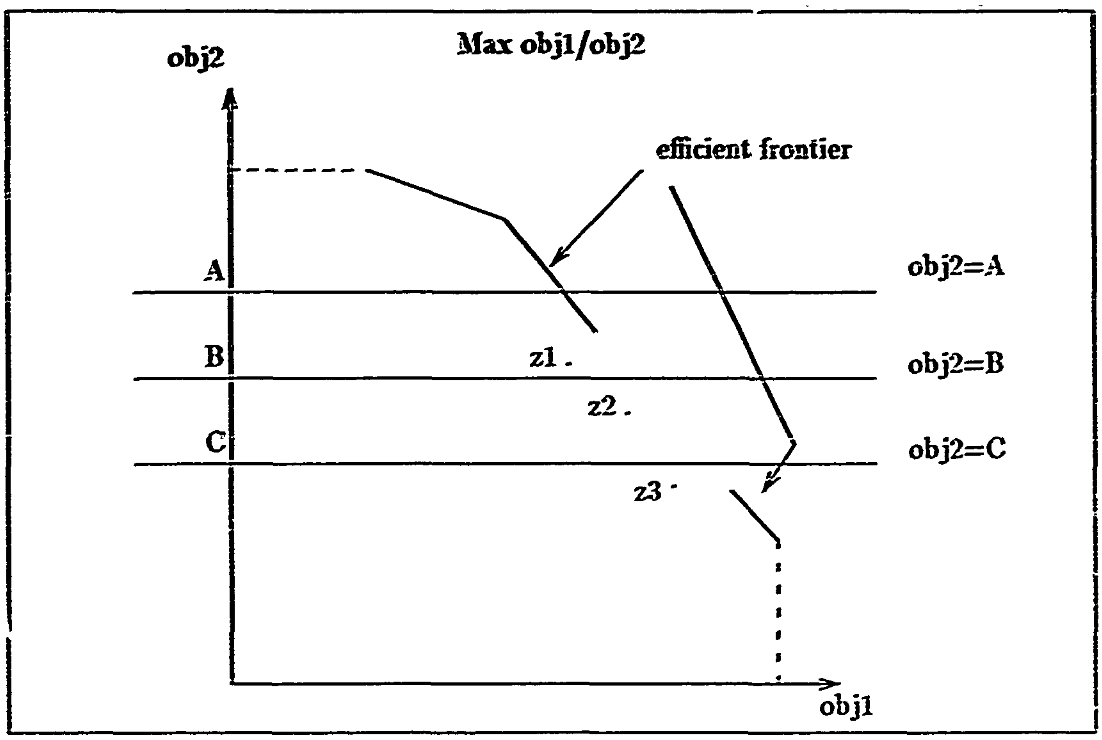
5.3.5 Correlation Between objectives. Complicating the use of the weighted sums approach is the degree to which the objectives are correlated. If the objective functions are correlated, the weighting vectors can behave inconsistently (23:198). In other words, an important objective that is highly correlated with another objective may have a very low weighting for the best solution which is counter-intuitive. An acceptable method for measuring the correlation between objectives is to measure the angle between objective cost vectors (23:198). The measure to which the ith and jth objectives are correlated can be calculated by the formula:

$$\alpha = \cos^{-1}\left(\frac{(c^i)^T c^j}{\| c^i \|_2 \| c^j \|_2}\right)$$

The smaller the angle α, the greater the correlation. The angle between objectives

one and two for the MOLIP formulation is 42.6 degrees for time block one data and 41.4 degrees for time block seven data. The correlation between these objectives is neither high nor low, since the angle is midway between zero and 90 degrees.

Figure 5. Illustration of Bounding Constraints for the MOLIP



5.3.6 Constraint Reduced Feasible Region Method. Constraint reduction of the feasible region is a method which attempts to *trap* an optimal solution by selecting one of the objectives for maximization subject to bounds on the other objectives thereby forming a new set of constraints (23:202). The resulting feasible region is a subset of the original feasible region which existed with multiobjectives. The GSARP feasible region can be reduced by forming a bounding constraint with the second objective function as shown in Figure 5. Constraining the second objective provides a means for investigating all possible integer levels of excess coverage, thus potentially revealing additional supported and unsupported pareto optimal integer solutions for the MOLIP.

5.3.7 Scaling the Objective Functions. "Three philosophies are available for rescaling the objective functions: normalization, use of 10 raised to an appropriate power, and the application of range equalization factors (23:200)." When the desire to use scaling is to bring all objective coefficients to the same order of magnitude, using 10 raised to an appropriate power is a viable alternative to normalization (23:200). One reason using an appropriate power of ten may be preferable to normalization is that the original coefficients are still recognizable, since they differ by only a decimal point. Range equalization is used if it is desired to equalize the ranges of possible objective function values . Due to reduced problem size, the test case results presented in chapter four used a scaled version of the first objective. Scaling is not needed for the full scale thesis problem, since both of the objectives are the same order of magnitude.

5.3.8 Analysis of the GSARP Efficient Frontier. For many multiobjective optimization problems, a decision maker is presented with a set of available options which are presented from the set of pareto optimal solutions. From these options, the decision maker, based on his judgement and knowledge of the problem, selects the pareto optimal point which he considers "best" . For the GSARP, there is only one true objective which is to maximize the number of expected geolocations in the SAR network. Therefore, the set of pareto optimal solutions identified by the MOLIP must be mapped to the solution space of the GSARP, which is referred to in terms of expected geolocations.

Evaluation of a GSARP solution using the full nonlinear objective requires that every combination of receiving stations be evaluated with every frequency and transmitter location combination, involving millions of calculations. The department of Defense has provided a program, EVAL, to evaluate any given allocation of HFDF bundles to stations for any given frequency assignment to the HFDF receivers in a SAR network(8). EVAL can be used to calculate the true objective function or it can be used to approximate the true objective function by evaluating just a subset of possible receiving station combinations for each transmitter location/frequency combination(14:32). EVAL is documented to provide good results when the number of stations considered for each transmitting location/frequency combination is between ten and twelve (8). Also, the increase in the

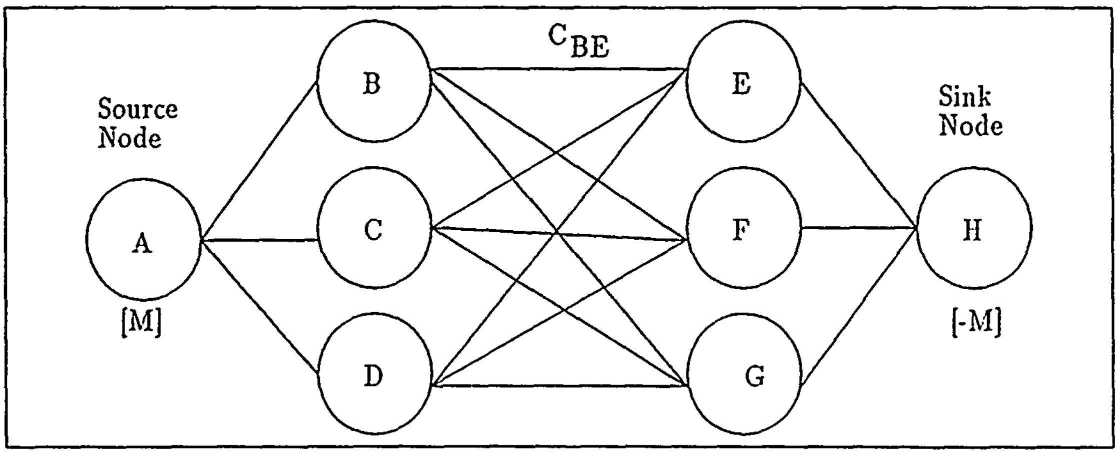
objective function when more than ten to twelve stations are considered diminishes as each additional station is added while the computation time required for the evaluation is doubled(14:32).

5.4 Network Representation

For many linear programming problems, there are advantages to using a network representation. The primary advantage for some linear programming problems is that special more efficient network algorithms can be used in place of linear programming. Some problems have a unimodular constraint matrix guaranteeing integer solutions without restricting the variables to be integer. Some problems can still benefit from using a network representation, although they may not have a unimodular constraint structure and cannot take advantage of efficient network algorithms.

A network representation can be used to better understand the flow of the problem and how the constraints are tied together. For some problems, such as the GSARP, a network representation makes it easier to identify arcs which must be explicitly integerized to guarantee an integer solution.

Figure 6. Example of Network Representation



In a network representation nodes and arcs are used to represent the flow of resources. The flow through the simple network in Figure 6 will be described to illustrate the use of the network representation. The nodes are labeled as A through H. Node A is referred to

as the source node and there are M units of resources that must flow from the source node directly into the network via arcs AB, AC, and AD. Conservation of flow must be present at every node in the network. In other words, the flow into a node must equal the flow out of a node. For example, the flow into AB must equal the sum of the flows BE, BF and BG. Node H is referred to as the sink node and M refers to the amount of flow that is required to flow into node H. The objective function is partially represented as the cost of flow C_{BE} multiplied by the amount of flow in arc BE. The sum of all such flows multiplied by their associated costs can be maximized or minimized.

A network can be effectively used to represent the GSARP. Two possible network representations are presented in detail in the next chapter, Network Representations for the MOLIP.

5.5 *Solution Strategy*

The integer requirements of the GSARP eliminate the use of pure linear programming or ADBASE. Within these constraints a solution strategy is outlined:

- Determine a network representation that is appropriate for the MOLIP.
- Determine which variables must be integerized to guarantee an integer solution.
- Using SAS LP with a limited number of integer variables, apply the weighted sums approach to determine the efficient frontier for two time blocks using two formulations: one with a covering constraint and one without a covering constraint.
- Use EVAL to evaluate the pareto-optimal extreme points identified by the weighted sums approach.
- Use the constraint reduced feasible region method to search for additional supported and unsupported pareto-optimal points, which correspond to areas of the efficient frontier that produce good EVAL results.
- Compare the best-pareto optimal solution identified from EVAL to the mean and standard deviation of EVAL results found by the Department of Defense for 1,000

randomly generated locations that are tasked by maximizing the expected number of lines of bearing for each station.

The solution strategy outlined above will be used to arrive at the results presented in chapter seven.

VI. Network Representations for the MOLIP

Two network representations are presented for the MOLIP. The first is a single-stage network which has two sets of integer variables. The second is a two-stage network which has one set of integer variables for each stage. Test case results show that a two-stage network produces solutions similar to a single-stage network, using significantly less computation time.

6.1 Notation

The following notation is used to describe the network representations presented in this chapter. The majority of the notation described in this section is for decision variables which are represented as the flow from one arc to another. In general, the first two capital letters represent the start node for the flow and the last two capital letters represent the end node for the flow. These arc flow decision variables correspond directly the the flow pictured in the network representations.

$$\mathrm{SOST}j = \begin{cases} 1 & \text{if a receiving station is located at } j \text{ in the single-stage} \\ & \text{network representation.} \\ 0 & \text{otherwise.} \end{cases}$$

$$\mathrm{SO1ST}j = \begin{cases} 1 & \text{if a receiving station is located at } j \text{ in stage one of} \\ & \text{the two-stage network representation.} \\ 0 & \text{otherwise.} \end{cases}$$

$$\mathrm{SO2ST}j = \begin{cases} 1 & \text{if two bundles are located at station } j \text{ in stage-two of} \\ & \text{the two-stage network representation.} \\ 0 & \text{otherwise (one bundle is located at station } j\text{).} \end{cases}$$

$$\mathrm{ST}j\mathrm{S1} = \begin{cases} 1 & \text{if receiving station } j \text{ is assigned one bundle of HFDF receivers in single-stage network representation.} \\ 0 & \text{otherwise (either receiving station } j \text{ is located and assigned two bundles of HFDF receivers or receiving station } j \text{ is not located).} \end{cases}$$

$$\mathrm{ST}j\mathrm{ST} = \begin{cases} 2 & \text{if receiving station } j \text{ is assigned two bundles of HFDF receivers in single-stage network representation.} \\ 1 & \text{if receiving station } j \text{ is assigned one bundle of HFDF receivers.} \\ 0 & \text{if receiving station } j \text{ is not located.} \end{cases}$$

$$\mathrm{ST}j\mathrm{F}k = \begin{cases} 1 & \text{if an HFDF receiver is located at station } j \text{ transmitting on frequency } k. \\ 0 & \text{otherwise.} \end{cases}$$

FkEX=arc flow from the frequency node k to the excess coverage node. This represents the units of excess coverage assigned to frequency k. Excess coverage is any coverage to a frequency beyond its fair share (FS).

FkNE=arc flow from the frequency node k to the nonexcess coverage node. This represents the units of coverage assigned to frequency k that are not excess coverage. These arcs have a capacity of eight.

EXSI =arc flow from excess coverage node to the termination node or sink. This represents the total units of excess coverage assigned to all frequencies.

NESI =arc flow from nonexcess coverage node to the termination node or sink. This represents the total number of units of excess coverage assigned to all frequencies.

G=the multiplicative gain (in flow) of an arc. If m units flow into the arc G× m units flow out of an arc.

{M}=external flow requirements at the source and sink. At the source M is positive and represents the required number of units that must flow out of the source. At the sink and slack M is negative and represents the required number of units that must flow into the sink.

J=the set of stations selected in stage one

NS=total number of stations to be located among j stations.

NB=total number of bundles of HFDF receivers to be allocated to NS stations.

FS=the fair share of HFDFs for any frequency. Fair share is defined as NS×G/K.

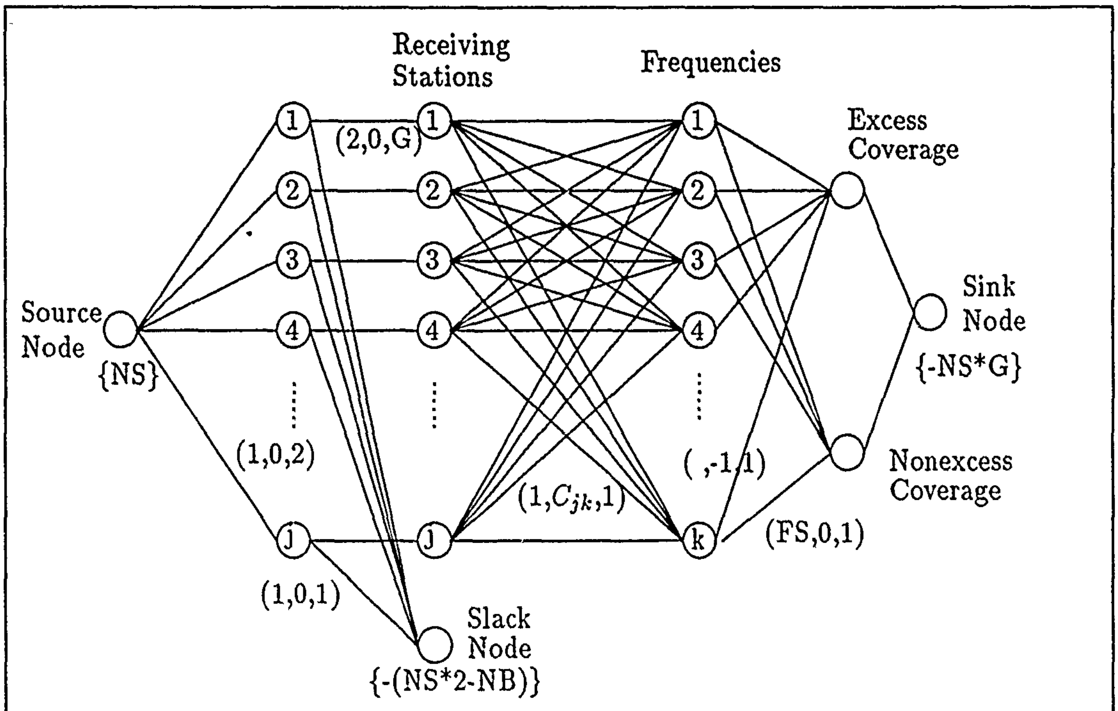
6.2 Single-Stage Network

The single-stage network representation in Figure 7 is a full representation of the MOLIP (discussed in Chapters 3 and 5) with no covering constraint used to replace the quasi-covering constraint discussed in 5.2.2. There are two sets of arcs that must be explicitly integerized in order to represent the MOLIP in a single network. The first set of arcs emanating from the source node must be integerized in order to select the stations. The slack arcs must also be integerized as these arcs regulate whether a station that is selected gets one or two bundles of HFDF receivers.

6.3 Mathematical Representation of Single-Stage Network

The single-stage mathematical formulation is primarily a set of conservation of flow equations corresponding to the nodes shown in the Figure 7. The source, slack, and sink

Figure 7. Single-Stage Network Representation



Legend: (arc capacity,cost,gains parameter)

NS = number of stations to be assigned

FS = fairshare of coverage for each frequency

NB = number frequency bundles to be assigned

G = gains parameter { } = fixed external flow

G*arcflow = the flow into the next node

nodes use their fixed external flows shown in brackets as part of their conservation of flow equations. This formulation corresponds closely to the MOLIP presented in chapter three.

$$\max \quad \sum_{j}^{J}\sum_{k}^{K} C_{jk}\mathrm{ST}j\mathrm{F}k$$

$$\min \quad \mathrm{EXSI}$$

where $C_{jk} = \sum_i^I W_{ij}P_{ijk}F_{jk}$
subject to

$$\begin{aligned}
\sum_{j}^{J}\mathrm{SOST}j &= \mathrm{NS} \\
2\times\mathrm{SOST}j - \mathrm{ST}j\mathrm{SL} - \mathrm{ST}j\mathrm{ST} &= 0, \quad \forall\, j \\
\sum_{j}^{J}\mathrm{ST}j\mathrm{SL} &= 2\times\mathrm{NS} - \mathrm{NB} \\
\mathrm{G}\times\mathrm{ST}j\mathrm{ST} - \sum_{k}^{K}\mathrm{ST}j\mathrm{F}k &= 0, \quad \forall\, j \\
\sum_{j}^{J}\mathrm{ST}j\mathrm{F}k - \mathrm{F}k\mathrm{EX} - \mathrm{F}k\mathrm{NE} &= 0, \quad \forall\, k \\
\sum_{k}^{K}\mathrm{F}k\mathrm{EX} - \mathrm{EXSI} &= 0 \\
\sum_{k}^{K}\mathrm{F}k\mathrm{NE} - \mathrm{NESI} &= 0 \\
\mathrm{EXSI} + \mathrm{NESI} &= (\mathrm{NS}\times\mathrm{G}) \\
\mathrm{ST}j\mathrm{F}k &\leq 1, \quad \forall j,\ k \\
\mathrm{F}k\mathrm{NE} &\leq FS, \quad \forall k \\
\mathrm{SOST}j &\in \{0,1\} \\
\mathrm{ST}j\mathrm{SL} &\in \{0,1\}
\end{aligned}$$

6.4 Computational Experience with the Single-Stage Network

To investigate the computational tractability of he single-stage network formulation, several runs for the full thesis problem were attempted using SAS LP. This formulation

of the full thesis problem has 1084 variables, 55 of which were restricted to be binary. Every run that was attempted terminated prematurely using all available storage space, after two to four CPU hours on a VAX 11/785. In each case, the program crashed because there was insufficient space to continue after filling up as much as 400 thousand blocks on some runs. The SAS work directories were using as excessive amount of space storing the branch-and-bound tree as it searched for an integer solution. Local computing platform limitations on the available memory and space for utility data sets prevented successful completion of any runs, rather than SAS-related limitations.

The single-stage network has proven to be computationally intractable due to computing resource limitations. An alternate formulation is desired which can provide comparable results while using fewer resources. Such an alternate formulation is presented in the next section.

6.5 Two-Stage Network

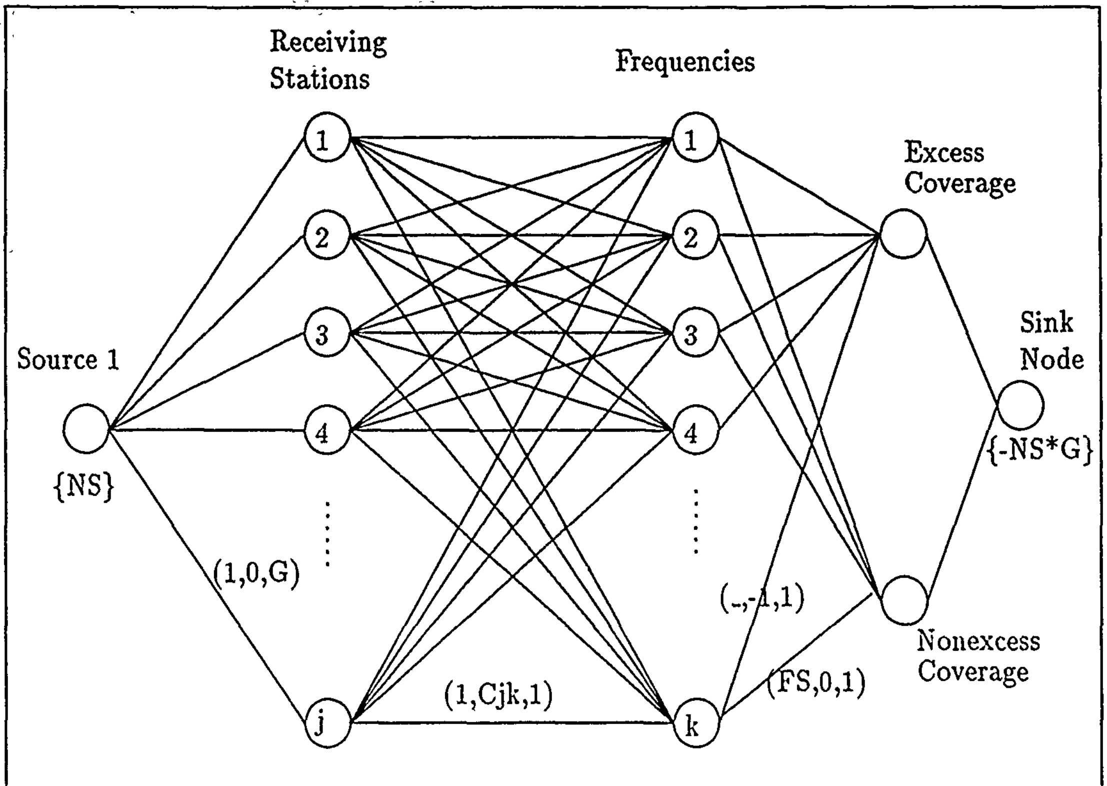
A formulation is discussed which uses fewer computer resources by significantly reducing the storage space needed for branch and bound tree . The strategy was to formulate the MOLIP using fewer explicitly defined binary variables. A two-stage network representation was investigated since it provided a means to reduce the number of integer arcs in half for each stage. The two-stage concept was motivated from a multicommodity flow formulation. That is to say, the first and second bundle of HFDFs received by a station are treated as two different commodities.

In the two-stage concept for the MOLIP, the first stage selects the stations to be located and the second stage selects the number of bundles each selected station receives. It is clear that the two-stage approach is an approximation to the single-stage. The remaining sections present the networks and mathematical representations of the two-stage concept. Results are also presented which document that the two-stage formulation performs similarly to the single-stage formulation on smaller problems.

6.5.1 Stage-One Representation. The first stage pictured in Figure 8 integerizes the set of arcs from source 1 to the stations so that the stations can be selected. Each

station selected will automatically be assigned one bundle of HFDF receivers in stage two.

Figure 8. Stage-One Network Representation



Legend: (arc capacity,cost,gain parameter)

NS = number of stations

FS = fair share of coverage for each frequency

G = gain G*arcflow = flow into next node

{ } = fixed external flow

6.5.2 Mathematical Representation of Stage One. The stage-one mathematical formulation is also primarily a set of flow conservation equations corresponding to the nodes shown in Figure 8. The source and sink nodes use the fixed external flows (shown in brackets) as part of the flow conservation equations.

$$\max \sum_{j}^{J}\sum_{k}^{K} C_{jk} \mathrm{ST}j\mathrm{F}k$$

$$\min \quad \mathrm{EXSI}$$

where $C_{jk} = \sum_{i}^{I} W_{ij} P_{ijk} F_{jk}$

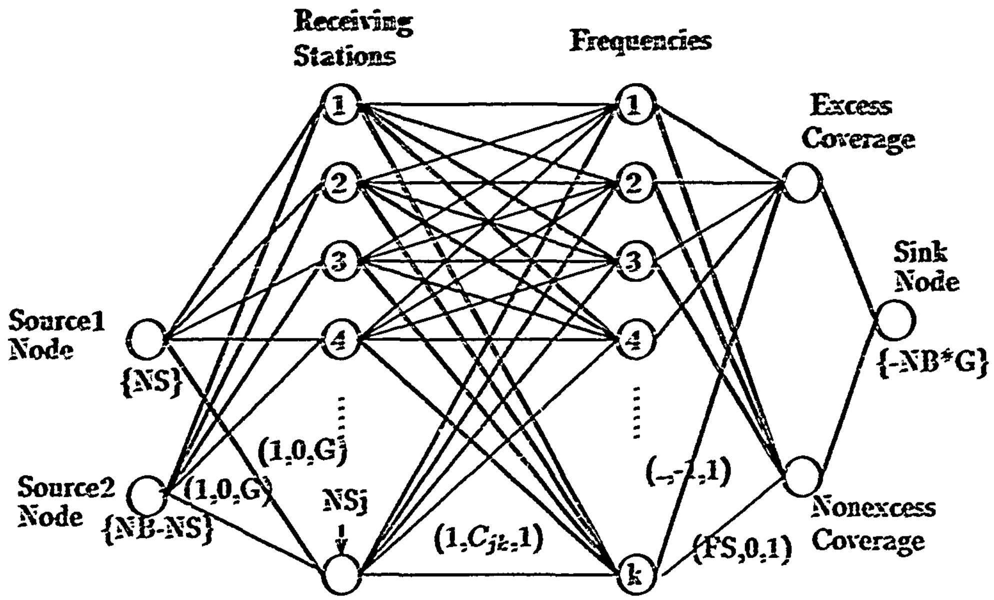
subject to

$$\begin{aligned}
\sum_{j}^{J} \mathrm{SO1ST}j &= \mathrm{NS} \\
\mathrm{G} \times \mathrm{SO1ST}j - \sum_{k}^{K} \mathrm{ST}j\mathrm{F}k &= 0, \quad \forall j \\
\sum_{j}^{J} \mathrm{ST}j\mathrm{F}k - \mathrm{F}k\mathrm{EX} - \mathrm{F}k\mathrm{NE} &= 0, \quad \forall k \\
\sum_{k}^{K} \mathrm{F}k\mathrm{EX} - \mathrm{EXSI} &= 0 \\
\sum_{k}^{K} \mathrm{F}k\mathrm{NE} - \mathrm{NESI} &= 0 \\
\mathrm{EXSI} + \mathrm{NESI} &= \mathrm{NS} \times \mathrm{G} \\
\mathrm{ST}j\mathrm{F}k &\leq 1, \quad \forall j, k \\
\mathrm{F}k\mathrm{NE} &\leq FS, \quad \forall k \\
\mathrm{SO1ST}j &\in \{0,1\}
\end{aligned}$$

6.5.3 Stage-Two Representation. In the second stage, the set of arcs from the second source to the stations selected in stage one are integerized to determine which stations are to receive a second bundle of HFDF receivers.

6.5.4 Mathematical Representation of Stage Two. The stage-two mathematical formulation is primarily a set of flow conservation equations corresponding to the nodes shown in the Figure 9. The source and sink nodes use fixed, external flows (shown in brackets) as part of flow conservation equations.

Figure 9. Stage-Two Network Representation



Legend: (arc capacity,cost,gain parameter)

NS = number of stations chosen in stage one
NB = total number of bundles to be assigned
FS = fairshare of coverage for each frequency
G = gains parameter   G*arcflow = flow into next node
{ } = fixed external flow

$$\max \quad \sum_{j}^{J}\sum_{k}^{K} C_{jk}\mathrm{ST}j\mathrm{F}k$$

$$\min \quad \mathrm{EXSI}$$

where $C_{jk} = \sum_{i}^{I} W_{ij} P_{ijk} F_{jk}$
and $J \in$ { **stations chosen in stage one** }
subject to

$$\begin{aligned}
\mathrm{SO1ST}j &= 1, \ \forall j \in J \\
\sum_{j}^{J}\mathrm{SO2ST}j &= (\mathrm{NB}-\mathrm{NS}) \\
\mathrm{G}\times\mathrm{SO1ST}j + \mathrm{G}\times\mathrm{SO2ST}j - \sum_{k}^{K}\mathrm{ST}j\mathrm{F}k &= 0, \ \ \forall j \\
\sum_{j}^{J}\mathrm{ST}j\mathrm{F}k - \mathrm{F}k\mathrm{EX} - \mathrm{F}k\mathrm{NE} &= 0, \ \ \forall k \\
\sum_{k}^{K}\mathrm{F}k\mathrm{EX} - \mathrm{EXSI} &= 0 \\
\sum_{k}^{K}\mathrm{F}k\mathrm{NE} - \mathrm{NESI} &= 0 \\
\mathrm{EXSI} + \mathrm{NESI} &= \mathrm{NS} * \mathrm{G} \\
\mathrm{ST}j\mathrm{F}k &\leq 1, \ \ \forall j, k \\
\mathrm{F}k\mathrm{NE} &\leq FS, \ \ \forall k \\
\mathrm{SO2ST}j &\in \{0,1\}
\end{aligned}$$

6.6 Computational Experience with Network Representations

Using the weighted sums approach, pareto-optimal solutions for two test cases were examined. The purpose of the case study was to determine if the two-stage network representation could provide comparable results to the single-stage network representation ,in addition to being more efficient. The test cases were designed with a scaled-down version of the thesis problem data. st fifteen of thirty receiving stations and all

thirty-one frequencies were used. Three of the receiving stations were fixed, representing a fixed, base network. Data for the respective case studies was selected from time blocks one and seven.

The single-stage network optimization selects ten receiving stations with five stations receiving one bundle of HFDF receivers and the other stations receiving two bundles. Unlike the single-stage network, the two-stage network does not attempt to optimize station and bundle location in one step. In the first stage, 10 stations are located . Each station from stage one is allocated one bundle of HFDF receivers. In the second stage, an additional five bundles of HFDF receivers are allocated to five of the stations located during stage one.

Table 3. Case 1: Comparison of Network Representations using Time Block One

Solution	Single-Stage EVAL	Two-Stage EVAL	% diff
λ_1=1.0	6.90	6.90	none
λ_1=0.99	6.90	6.90	none
λ_1=0.975	6.90	6.90	none
λ_1=0.95	6.90	6.90	none
λ_1=0.90	6.67	6.67	none
λ_1=0.85	6.45	6.45	none
λ_1=0.80	6.37	6.37	none
λ_1=0.75	5.64	5.64	none
λ_1=0.70	4.25	4.93	16%
λ_1=0.65	3.81	3.57	-6%
$\lambda_1 \leq$0.60	3.81	2.94	-22.8%
average computation time	17 min 51 sec	1 min 29 sec	-91%

Test case results shown in Figure 3 and Figure 4 illustrate that the total CPU time required by the two-stage network is significantly less than for the single-stage network. The required CPU time is reduced by more than 90% in both test cases. Furthermore, both network representations produced similar results for both network configurations and EVAL results. These findings were consistent for both test cases.

Table 4. Case 2: Comparison of Network Representations using Time Block Seven

Solution	Single-Stage EVAL	Two-Stage EVAL	% diff
λ_1=1.0	5.16	5.16	none
λ_1=0.99	5.16	5.16	none
λ_1=0.975	5.16	5.16	none
λ_1=0.95	5.12	5.12	none
λ_1=0.90	4.81	4.81	none
λ_1=0.85	4.48	4.48	none
λ_1=0.80	4.45	4.45	none
λ_1=0.75	3.98	3.98	none
λ_1=0.70	3.54	3.54	none
λ_1=0.65	3.73	3.04	-18%
$\lambda_1 \leq$0.60	3.73	3.02	-19%
average computation time	26 min 59 sec	1 min 3 sec	-96%

6.7 Solution Strategy Revisited

Chapter five presented an initial solution strategy. This chapter addressed the first two items mentioned in that strategy which are, restating:

- Determine a network representation that is appropriate for the MOLIP.
- Determine which variables must be integerized to guarantee an integer solution.

A two-stage network representation evolved to reduce the computing time and resources needed to solve the MOLIP formulation. Computational experience with the two-stage MOLIP formulation, which uses half the number of integer variables during each stage, shows a significant reduction can be achieved in computation time and resources. Furthermore, case study results show the two-stage results are comparable to the single-stage for both test cases. Consequently, the two-stage formulation was determined to be appropriate for the computation of thesis results presented in the next chapter. Chapter 7 will present results obtained by applying the two-stage MOLIP with the solution strategy outlined in Chapter 5.

VII. Results, Conclusions and Recommendations

This chapter summarizes and discusses the results obtained from the solution strategies presented in Chapters 5 and 6. The Department of Defense provided all necessary transmission and reception probabilities for the twelve two-hour time blocks. Results are presented for two time blocks to validate that the two-stage heuristic works well with contrasting sets of data. Specifically, time block one and seven, which are separated by ten hours, are used. For each time block, results with and without a covering constraint are compared.

7.1 Time Block One Results

The results for time block one can be found in table 5 and table 6. These tables show results from both the weighted sums approach and the constraint reduced feasible region method. Results from the weighted sums approach can be identified by a λ_1 value

Table 5. Results for Time Block One Without Covering Constraint

Solution	Station Config	Bundle Config	Objective 1	Objective 2	EVAL result
λ_1=1.0	Config A	Config I	136.60	-97	17.345
λ_1=0.99	A	I	136.59	-96	17.340
λ_1=0.975	A	I	136.59	-96	17.340
λ_1=0.95	A	I	136.57	-95	17.340
Obj2=-94	A	I	136.52	-94	17.196
Obj2=-92	A	I	136.38	-92	17.057
Obj2=-90	A	I	136.23	-90	17.047
Obj2=-88	A	I	136.03	-88	16.910
λ_1=0.90	A	I	135.82	-86	16.877
Obj2=-85	A	I	135.70	-85	16.900
λ_1=0.85	A	Config II	134.25	-75	16.530
λ_1=0.80	A	II	131.96	-64	15.875
λ_1=0.75	A	Config III	131.71	-54	15.477
λ_1=0.70	Config B	Config IV	119.37	-30	14.580
λ_1=0.65	B	Config V	110.05	-11	12.412
λ_1=0.60	B	V	104.60	-2	12.010
λ_1 <=0.55	B	V	104.60	0	11.700
Specific taskings for each solution are on floppy disk 2 in directory *time1.*					

in the solution column, whereas results from the constraint reduced feasible region method

list an *Obj2* value in the solution column. The best EVAL solution of 17.345 was found using the formulation without a covering constraint with 100% weight on the first objective and 0% weight on the second objective. The best result with a covering constraint was 16.66. Several configurations for stations and bundles are generated by the MOLIP.

Table 6. Results for Time Block One With a Covering Constraint

Solution	Station Config	Bundle Config	Objective 1	Objective 2	EVAL result
λ_1=1.0	Config A	Config I	128.87	-75	16.36
λ_1=0.99	A	II	128.86	-72	16.54
Obj2=-70	A	II	128.83	-70	15.99
λ_1=0.975	A	II	128.80	-69	**16.66**
λ_1=0.95	A	III	128.69	-66	16.57
Obj2=-63	A	Config IV	128.31	-63	16.11
Obj2=-61	A	Config IV	128.17	-61	16.15
λ_1=0.90	A	Config IV	128.07	-60	16.13
λ_1=0.85	A	Config IV	127.32	-55	16.04
λ_1=0.80	A	Config IV	126.11	-49	15.42
λ_1=0.75	Config B	Config V	121.47	-36	15.06
λ_1=0.70	B	V	119.19	-30	14.50
$\lambda_1 \leq$=0.65	B	V	104.11	0	12.71
Specific taskings for each solution are on floppy disk 2 in directory *cover1.*					

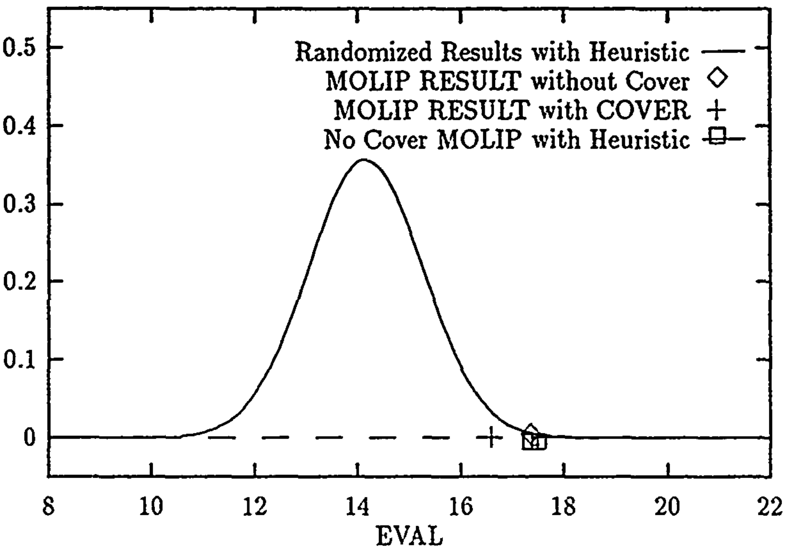
For time block one, the formulation without a covering constraint generates five unique station/bundle configurations which can be mapped to just two station configurations. Several pareto-optimal solutions can be mapped to a single configuration. For example, station configuration A and bundle configuration I are identical for the first 10 pareto-optimal solutions in table 5. These solutions differ only by their frequency taskings. A similar set of unique station/bundle configurations exists for the *covering* formulation.

On a VAX 11/785, the average CPU time required for one pareto-optimal solution was approximately 10 minutes for the formulation without a covering constraint and 37 minutes for the formulation with a covering constraint. The second stage of the optimization accounted for approximately 90% of this CPU time.

The Department of Defense provided comparison results for a randomized set of locations tasked with a heuristic that maximizes the lines of bearing individually at each station. While not optimal, this heuristic on the average provides good feasible taskings

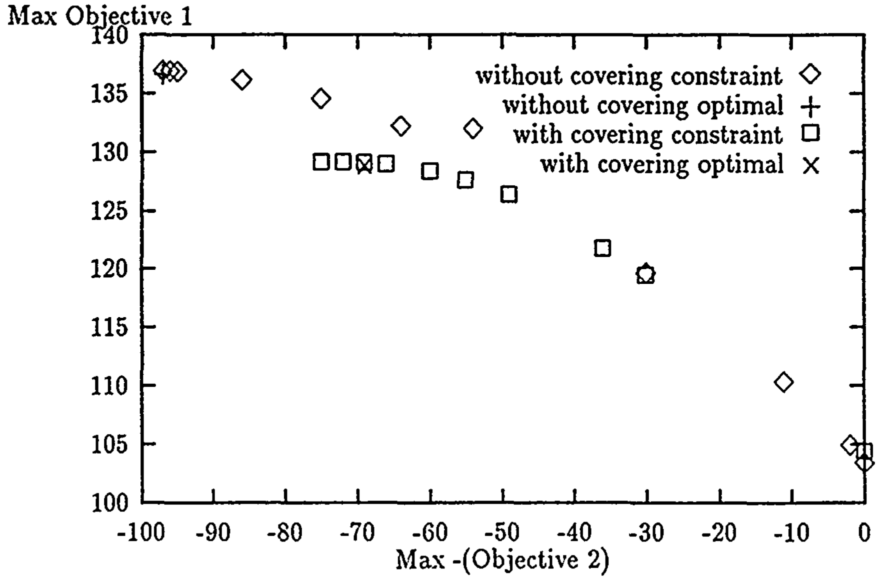
for a fixed network which are better than a randomized tasking(10). For time block one, a standard normal curve representing these comparison results is compared to the MOLIP results in Figure 10. These results had a mean of 14.149 and a standard deviation of

Figure 10. Time Block One Comparison of Results



1.1175 for 3500 random samples , and a high and low of 17.71 and 10.321 respectively. The best MOLIP solution for the first time period was 2.86 standard deviations above the mean of randomized locations that are heuristically tasked. Furthermore, when the best MOLIP configuration was tasked with the same heuristic as the randomized locations it produced an EVAL solution of 17.5145 which is 3.01 standard deviation above the mean of the randomized locations. These results show that with a maximum lines of bearing heuristic, the MOLIP was able to provide a good feasible configuration that is in most cases better than any of the randomized configurations . Figure 11 shows a plot of both efficient frontiers for the first time block. The second objective function value, z_2, is plotted on the x-axis and the first objective function value, z_1 is plotted on the y-axis for each pareto-optimal solution. The shape of the efficient frontier is similar both with and without the covering constraint. The efficient frontier of the covering formulation lies inside the convex hull of solutions for the *no cover* formulation. Symbols with + or x inside them represent the pareto-optimal solutions which correspond to the best EVAL solution. In general, the EVAL solutions corresponding to MOLIP solutions are unimodal. As a result,

Figure 11. Efficient Frontiers for Time Block One



the constraint-reduced feasible region method was only used to generate additional pareto-optimal solutions in the neighborhood of the best EVAL solution. The additional solutions provided by the constraint reduced feasible region method did not unveil any unusual or unexpected information about the MOLIP's efficient frontier or the MOLIP's best EVAL solution.

7.2 Time Block Seven Results

Table 7 and table 8 have the results for time block seven with both the weighted sums approach and the constraint reduced feasible region method. For time block seven, the formulation without a covering constraint generates four unique station/bundle configurations which can be mapped to four configurations of stations. As with time block one, several pareto-optimal solutions can be mapped to a single configuration. For example, the first three pareto-optimal solutions in table 7 all have station configuration A and bundle configuration I which differ only by their frequency taskings. For the *covering* formulation, the six bundle configurations are mapped to the same set of stations.

On a VAX 11/785, the average CPU time required for one pareto optimal solution of time block seven was approximately 6 minutes for both formulations. The second stage

Table 7. Results for Time Block Seven Without Covering Constraint

Solution	Station Config	Bundle Config	Objective 1	Objective 2	EVAL result
λ_1=1.0	Config A	Config I	111.27	-98	14.11
λ_1=0.99	A	I	111.27	-98	14.11
λ_1=0.975	A	I	111.27	-98	14.11
λ_1=0.95	Config B	Config II	110.75	-88	14.23
Obj2=-81	B	II	110.15	-81	14.47
λ_1=0.90	B	II	110.04	-80	14.40
Obj2=-77	B	II	109.68	-77	14.26
Obj2=-73	Config C	Config III	109.06	-73	14.47
Obj2=-71	C	III	108.78	-71	14.40
Obj2=-69	C	III	108.45	-69	14.71
Obj2=-68	C	III	108.23	-68	**14.72**
λ_1=0.85	C	III	108.11	-67	14.71
Obj2=-66	C	III	107.93	-66	14.69
Obj2=-64	C	III	107.56	-64	14.65
λ_1=0.80	C	III	104.47	-50	14.13
λ_1=0.75	Config D	Config IV	97.39	-26	13.64
λ_1=0.70	D	IV	92.02	-12	13.31
λ_1 $\leq$=0.65	D	IV	86.44	0	12.12
Specific taskings for each solution are on floppy disk 2 in directory *time7*.					

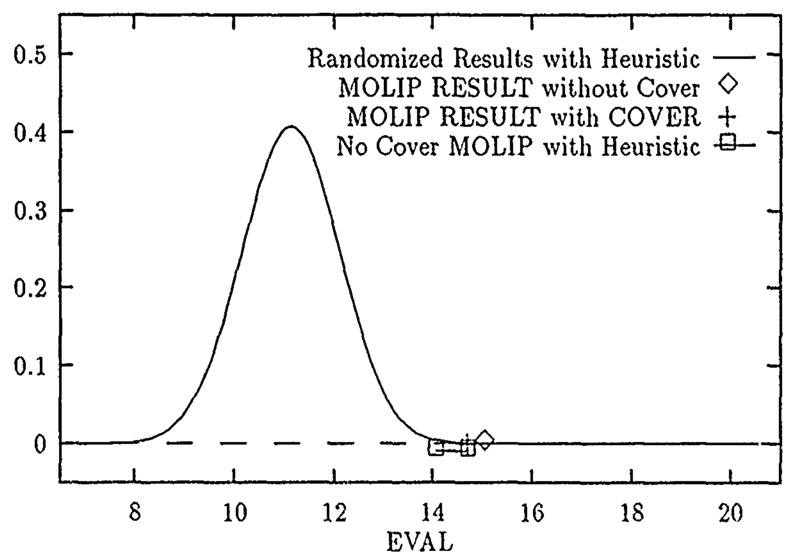
of the optimization accounts for approximately 80% of the total CPU time.

Unlike time period one results, the best EVAL solution of 15.06 was found using the formulation with a covering constraint. The best solution has a λ weight for the first objective in the range of 0.90 to 0.95. The best result without a covering constraint was of similar quality, with an EVAL result of 14.72. The Department of Defense provided similar comparison results for time period seven. A standard normal curve representing these comparison results is compared to the MOLIP results in Figure 12. The EVAL comparison results had a mean of 11.134 and a standard deviation of 0.9663 for 1000 randomized locations. The sample also had high and low EVAL values of 13.81 and 7.83 respectively. The performance of the best MOLIP solution for this time period exceeds the best comparison result and is more than four standard deviations above the mean of the randomized locations which were heuristically tasked. The best result with a covering constraint was evaluated with the maximum lines of bearing heuristic that was used to evaluate the randomized configurations. This produced an EVAL result of 14.07. This

Table 8. Results for Time Block Seven With a Covering Constraint

Solution	Station Config	Bundle Config	Objective 1	Objective 2	EVAL result
λ_1=1.0	Config A	Config I	105.98	-88	14.17
λ_1=0.99	A	I	105.98	-74	14.72
λ_1=0.975	A	I	105.97	-72	14.76
Obj2=-70	A	I	105.87	-70	14.88
Obj2=-69	A	Config II	105.83	-69	14.95
λ_1=0.95	A	II	105.78	-68	14.95
Obj2=-67	A	II	105.71	-67	14.96
Obj2=-66	A	II	105.63	-66	15.00
Obj2=-65	A	II	105.55	-65	**15.06**
Obj2=-64	A	II	105.43	-64	15.01
Obj2=-63	A	II	105.35	-63	14.98
λ_1=0.90	A	II	105.13	-61	14.99
λ_1=0.85	A	Config III	104.10	-54	14.19
λ_1=0.80	A	Config IV	102.56	-47	14.29
λ_1=0.75	A	Config V	96.99	-28	13.54
λ_1=0.70	A	Config VI	92.01	-15	12.84
$\lambda_1 \leq$=0.65	A	VI	84.68	0	12.27
Specific taskings for each solution are on floppy disk 2 in directory *cover7*.					

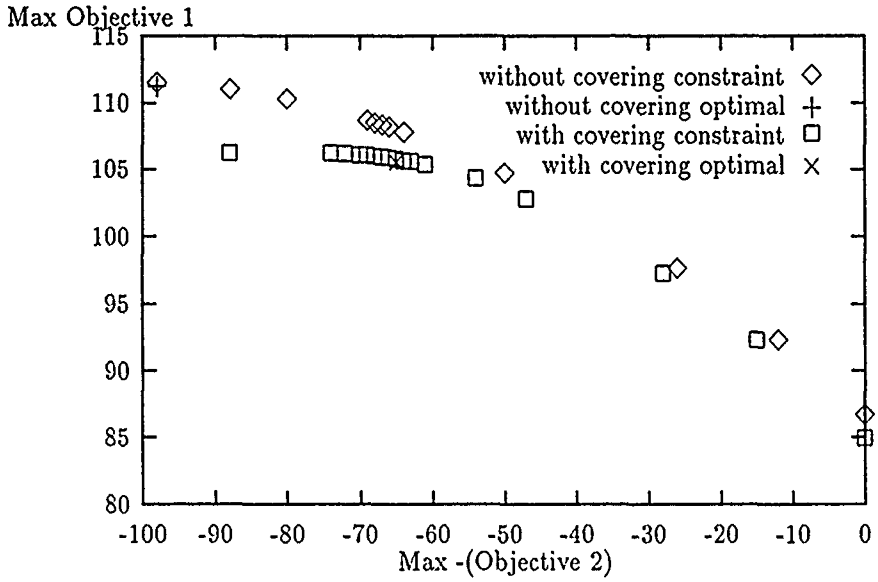
Figure 12. Time Block Seven Comparison of Results



result surpasses the maximum EVAL result found in any of the randomized locations, although it is still less than the best result the MOLIP found with either formulation These results confirm that with a maximum lines of bearing heuristic, the MOLIP can provide a good feasible configuration that is in most cases better than any of the randomized configurations.

Figure 13 shows a plot of both efficient frontiers for the seventh time block. The

Figure 13. Efficient Frontiers for Time Block Seven



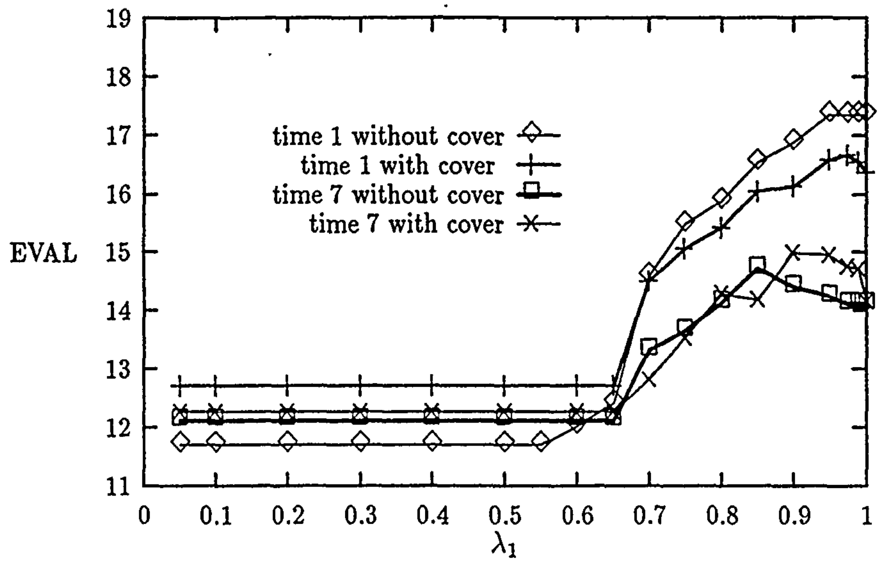
shape of efficient frontiers for this time block is similar to efficient frontiers shown in figure 11 for the first time block. The best EVAL solution was found using the covering formulation. As in the first time block, the efficient frontier of pareto-optimal solutions with the covering constraint lies inside the convex hull of pareto-optimal solutions without the covering constraint. Once again, nothing unusual or unexpected is identified about the MOLIP efficient frontier from the additional solutions provided by the constraint reduced feasible region method.

7.3 *Analysis of Results*

After analysis, several observations can be made about the MOLIP solutions for the GSARP.

- The two-stage MOLIP solution methodology is capable of consistently producing good feasible solutions for the GSARP. This result is supported by the comparison of MOLIP results to those generated by the Department of Defense for comparison. For time blocks one and seven, the MOLIP identified solutions that are 2.86 and 4.06 standard deviations, respectively, above the mean of randomly generated locations which are tasked by a greedy heuristic that provides good feasible solutions.
- The search space of the weighted sums approach can be reduced. For the contrasting time blocks, the best MOLIP solutions correspond to a value of λ_1 between 0.85 and 1.0 . Figure 14 illustrates that the EVAL results corresponding to the MOLIP solutions are approximately unimodal, and, in all cases, the maximum value occurs in the λ_1 range of 0.85 to 1.0. In fact, the optimal range for λ_1 has decreased from 0.383 to 1.0 in the very small test cases in Chapter 4, to this reduced range of 0.85 to 1.0 for the larger research problem. This might indicate that the weighting on objective one approaches unity as the problem size increases.
- The constraint-reduced feasible region method did not uncover unexpected or significantly improved results. That is to say, the character of the efficient frontier and corresponding EVAL solutions was sufficiently exposed during the weighted sums approach.
- The set of pareto-optimal solutions can be mapped to a set of station/bundle configurations. In many cases, several pareto-optimal solutions, which differ only by their frequency taskings, can be mapped to the same station/bundle configuration.
- The covering formulation performs well on the average compared to the no covering formulation. For time block one, the *covering* formulation's best result was 3.9% worse than the result with no covering constraint. On the other hand, the time block seven covering formulation produced a result that was 2.3% better than the result produced without a covering constraint. When considering just a single time block, neither formulation is clearly superior. However, the covering formulation is robust and makes more sense if multiple time periods are being simultaneously optimized, since it covers all the frequencies.

Figure 14. Comparison of EVAL Results



The MOLIP objective function can be generalized using the inequality $0.85 \leq \lambda_1 \leq 1.0$ where $\lambda_1 + \lambda_2 = 1$, to reduce the weighted sums search region. Figure 15 demonstrates the reduced search space resulting from a limited range for the λ weights.

$$\max \quad \lambda_1 z_1 + \lambda_2 z_2.$$

By defining a constant P, the optimal MOLIP solutions can be related to the EVAL solutions by the relationship:

$$P \times (\lambda_1 z_1 + \lambda_2 z_2) = \text{EVAL}.$$

Table 9 demonstrates, based on the research results presented for time blocks one and seven, how the value of P can be bounded for various ranges of λ_1 .

7.4 Conclusions

This research demonstrates that the MOLIP heuristic is a practical alternative to the proposed nonlinear formulation which is computationally intractable for realistic size problems. The MOLIP heuristic is robust, providing good feasible network configurations

Figure 15. Reduced Criterion Search Space

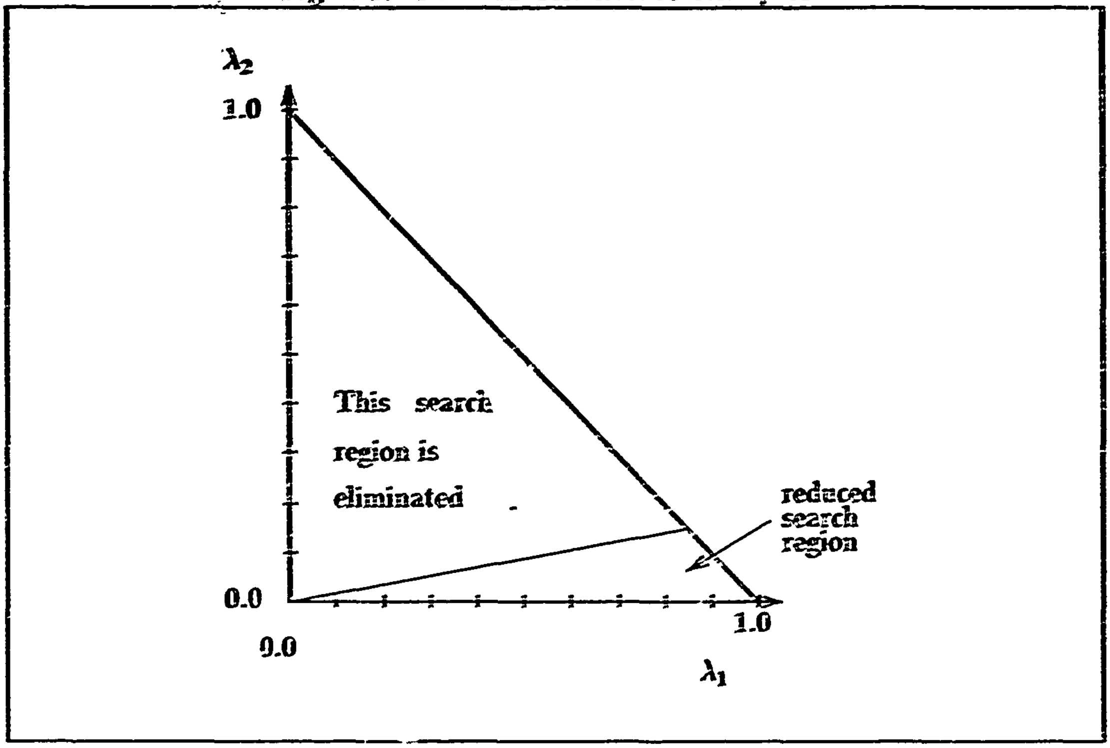


Table 9. λ_1 and P Value Ranges Relating the MOLIP and EVAL Results

λ_1 Range	Bounded P Range
$0.95 \leq \lambda_1 \leq 1.00$	$6.49 \leq P \leq 7.89$
$0.90 \leq \lambda_1 \leq 0.95$	$5.91 \leq P \leq 7.21$
$0.85 \leq \lambda_1 \leq 0.90$	$5.56 \leq P \leq 6.77$

for contrasting time periods, using similar weightings for the objectives in each case. The results presented herein indicate that the optimal λ_1 weighting lies in the range of 0.85 to 1.0. If the search region is reduced t is optimal range, the two-stage weighted sums MOLIP methodology can be used to efficiently determine good feasible SAR network configurations.

7.5 *Recommendations for Future Research*

Logical extensions for future research are :

1. Explore additional time periods to confirm the consistency of the two-stage MOLIP results and the consistency of the range of objective weightings.
2. Develop a modified MOLIP to allow more than just one or two bundles of HFDFs to be assigned to each station, that is, also allow for the possibility that some stations are assigned three or four bundles of HFDFs.
3. Apply the weighted sums two-stage network methodology to the comprehensive multi-time period GSARP discussed in Chapter 3. The multi-time period GSARP simultaneously solves the 12 time blocks to find one station/bundle configuration that is optimal. Using the reduced λ_1 search region from 0.85 to 1.0, determine the network configuration which simultaneously produces the best EVAL results for all of the time periods. Figure 16 in Appendix II illustrates this concept.
4. Bound the true optimal solution to the GSARP so that the quality of heuristic solutions can be compared to a common bound.

Appendix A. *Data Sorting*

Data for twelve time periods was provided by the Department of Defense. This FORTRAN program sorts the data and writes the data for each time period to a different file. A copy of this code called *append1* is provided on floppy disk 1 in the FORTRAN directory.

```
PROGRAM DATASPLIT

CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C                                                                   C
C PROGRAM DATASPLIT ADAPTED FROM CAPT KRISTA JOHNSON'S DATA SPLIT    C
C                                                                   C
C PROGRAM DATASPLIT READS TRANSMISSION PROBABILITIES FROM A FILE     C
C CALLED TGTPTX.DAT AND HFDF PROPAGATION PROBABILITIES FROM A FILE  C
C CALLED HFDFPAQ.DAT.  THE PROBABILITIES ARE WRITTEN TO SEPARATE     C
C ACCORDING TO TIME PERIOD.                                          C
C                                                                   C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC

      DIMENSION DP(40,30,31), F(40,31)

      OPEN (11,FILE='HFDFPAQ.DAT',STATUS='OLD')
      OPEN (12,FILE='TGTPTX.DAT',STATUS='OLD')

      OPEN (21,FILE='D1.DAT',STATUS='NEW')
      OPEN (22,FILE='F1.DAT',STATUS='NEW')

      OPEN (26,FILE='D2.DAT',STATUS='NEW')
      OPEN (27,FILE='F2.DAT',STATUS='NEW')

      OPEN (31,FILE='D3.DAT',STATUS='NEW')
      OPEN (32,FILE='F3.DAT',STATUS='NEW')

      OPEN (36,FILE='D4.DAT',STATUS='NEW')
      OPEN (37,FILE='F4.DAT',STATUS='NEW')
```

```
      OPEN (41,FILE='D5.DAT',STATUS='NEW')
      OPEN (42,FILE='F5.DAT',STATUS='NEW')

      OPEN (46,FILE='D6.DAT',STATUS='NEW')
      OPEN (47,FILE='F6.DAT',STATUS='NEW')

      OPEN (51,FILE='D7.DAT',STATUS='NEW')
      OPEN (52,FILE='F7.DAT',STATUS='NEW')

      OPEN (56,FILE='D8.DAT',STATUS='NEW')
      OPEN (57,FILE='F8.DAT',STATUS='NEW')

      OPEN (61,FILE='D9.DAT',STATUS='NEW')
      OPEN (62,FILE='F9.DAT',STATUS='NEW')

      OPEN (66,FILE='D10.DAT',STATUS='NEW')
      OPEN (67,FILE='F10.DAT',STATUS='NEW')

      OPEN (71,FILE='D11.DAT',STATUS='NEW')
      OPEN (72,FILE='F11.DAT',STATUS='NEW')

      OPEN (76,FILE='D12.DAT',STATUS='NEW')
      OPEN (77,FILE='F12.DAT',STATUS='NEW')

CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C                                                                    C
C THIS SECTION READS HFDF PROPAGATION PROBABILITIES FOR THE HFDF      C
C RECEIVERS AND WRITES THEM TO FILES ACCORDING TO TIME PERIOD         C
C                                                                    C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
```

```
      DO 100 I=1,40
         DO 80 J=1,30
            READ (11,'(A5)') STRIP1
            READ (11,'(A5)') STRIP2
            READ (11,900) (DP(I,J,K), K=1,31)
            WRITE (21,910) (DP(I,J,K), K=1,31)
            READ (11,900) (DP(I,J,K), K=1,31)
            WRITE (26,910) (DP(I,J,K), K=1,31)
            READ (11,900) (DP(I,J,K), K=1,31)
            WRITE (31,910) (DP(I,J,K), K=1,31)
            READ (11,900) (DP(I,J,K), K=1,31)
            WRITE (36,910) (DP(I,J,K), K=1,31)
            READ (11,900) (DP(I,J,K), K=1,31)
            WRITE (41,910) (DP(I,J,K), K=1,31)
            READ (11,900) (DP(I,J,K), K=1,31)
            WRITE (46,910) (DP(I,J,K), K=1,31)
            READ (11,900) (DP(I,J,K), K=1,31)
            WRITE (51,910) (DP(I,J,K), K=1,31)
            READ (11,900) (DP(I,J,K), K=1,31)
            WRITE (56,910) (DP(I,J,K), K=1,31)
            READ (11,900) (DP(I,J,K), K=1,31)
            WRITE (61,910) (DP(I,J,K), K=1,31)
            READ (11,900) (DP(I,J,K), K=1,31)
            WRITE (66,910) (DP(I,J,K), K=1,31)
            READ (11,900) (DP(I,J,K), K=1,31)
            WRITE (71,910) (DP(I,J,K), K=1,31)
            READ (11,900) (DP(I,J,K), K=1,31)
            WRITE (76,910) (DP(I,J,K), K=1,31)
80       CONTINUE
```

```
 100 CONTINUE

CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C                                                                  C
C THIS SECTION READS TRANSMISSION PROBABILITIES FOR EACH FREQUENCY C
C AT EACH SITE AND WRITES THEM TO FILES ACCORDING TO TIME PERIOD   C
C                                                                  C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC

      DO 200 I=1,40
        READ (12,'(A5)') STRIP1
        READ (12,'(A5)') STRIP2
        READ (12,900) (F(I,K), K=1,31)
        WRITE (22,910) (F(I,K), K=1,31)
        READ (12,900) (F(I,K), K=1,31)
        WRITE (27,910) (F(I,K), K=1,31)
        READ (12,900) (F(I,K), K=1,31)
        WRITE (32,910) (F(I,K), K=1,31)
        READ (12,900) (F(I,K), K=1,31)
        WRITE (37,910) (F(I,K), K=1,31)
        READ (12,900) (F(I,K), K=1,31)
        WRITE (42,910) (F(I,K), K=1,31)
        READ (12,900) (F(I,K), K=1,31)
        WRITE (47,910) (F(I,K), K=1,31)
        READ (12,900) (F(I,K), K=1,31)
        WRITE (52,910) (F(I,K), K=1,31)
        READ (12,900) (F(I,K), K=1,31)
        WRITE (57,910) (F(I,K), K=1,31)
        READ (12,900) (F(I,K), K=1,31)
        WRITE (62,910) (F(I,K), K=1,31)
```

```
      READ (12,900) (F(I,K), K=1,31)
      WRITE (67,910) (F(I,K), K=1,31)
      READ (12,900) (F(I,K), K=1,31)
      WRITE (72,910) (F(I,K), K=1,31)
      READ (12,900) (F(I,K), K=1,31)
      WRITE (77,910) (F(I,K), K=1,31)
200 CONTINUE

900 FORMAT (1X,31(F3.2,2X))
910 FORMAT (1X,31(F3.2,1X))

    END
```

Appendix B. *Computing Objective Function Coefficients*

Objective function coefficients are needed for SAS LP input files. This FORTRAN program calculates coefficients for the first objective function of any time period. A copy of this code called *append2* is provided on floppy disk 1 in the FORTRAN directory.

```
      PROGRAM OBJFUNC

CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C                                                                      C
C THIS PROGRAM COMPUTES OBJECTIVE FUNCTION COEFFICIENTS.  THE COEF-    C
C FICIENTS FOR THE OBJECTIVE FUNCTION ARE WRITTEN TO A FILE CALLED     C
C OBJT*.DAT.  WHERE * STANDS FOR THE TIME BLOCK.                       C
C THIS FILE HAS THE STATION NUMBER , THE FREQUENCY NUMBER AND THE      C
C CORRESPONDING OBJ FUNCTION COEFFICIENT IN EACH ROW.                  C
C                                                                      C
C D* and F* are the data files created by the data split program       C
C for time period *.  Ratioout.dat contains the weighting function     C
C function data Wij which is used for objective function one.          C
C                                                                      C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC

      DIMENSION F(40,31), DP(40,30,31), FAN(40,30), COEF(30,31)

      OPEN(10,FILE='D7.DAT',STATUS='OLD')
      OPEN(11,FILE='F7.DAT',STATUS='OLD')
      OPEN(12,FILE='RATIOOUT.DAT',STATUS='OLD')
      OPEN(13,FILE='OBJT7.DAT',STATUS='NEW')

      DO 100 I=1,40
        READ('',300) (F(I,K),K=1,31)
        DO 80 J=1,30
          READ(10,900) (DP(I,J,K),K=1,31)
          READ(12,910) FAN(I,J)
   80   CONTINUE
  100 CONTINUE
```

```
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C COMPUTE COEFFICIENTS FOR OBJECTIVE FUNCTION  C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC

      DO 200 J=1,30
        DO 180 K=1,31
          COEF(J,K) = 0.0
          DO 160 I=1,40
            COEF(J,K) = COEF(J,K) + F(I,K)*FAN(I,J)*DP(I,J,K)
  160     CONTINUE WRITE(13,920) J,K,COEF(J,K)
  180   CONTINUE
  200 CONTINUE


  900 FORMAT(1X,31(F3.2,1X))
  910 FORMAT(20X,F10.4)
  920 FORMAT(1X,I2,2X,I2,2X,F14.7)

      END
```

Appendix C. *Single-Stage SAS LP Input File*

This FORTRAN program generates the SAS LP input files for a single–stage network. It was specifically set up to generate input files for the test cases in Chapter 4. A copy of this code called *append*3 can be found on floppy disk 1 in the FORTRAN directory.

```
      PROGRAM SINGLESTAGE

CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C PURPOSE: THIS PROGRAM WRITES THE INPUT FILE FOR SAS PROC LP  C
C THE PROBLEM IS MULTIOBJECTIVE AND THEREFORE SEVERAL WEIGHT-  C
C INGS OF THE OBJECTIVE FUNCTION WILL BE INVESTIGATED.         C
C TOYWT.DAT       HAS THE OBJECTIVE FUNCTION WEIGHTINGS        C
C OBJT*.DAT       HAS THE OBJECTIVE FUNCTION COEFFICIENTS FOR  C
C                 TIME PERIOD '*' FOR OBJECTIVE ONE, WITH NO   C
C                 WEIGHTINGS APPLIED.                          C
C SL*W&n.SAS      IS THE SAS INPUT FILE FOR TIME PERIOD '*'    C
C                 AND WEIGHT '&'                               C
C Integer Variables (all binary) 12 SOSTj variables            C
C                         and     15 STjS variables            C
C This code generates test problems used in chapter six to     C
C validate the performance of a two-stage network against this C
C single-stage network.  Only 15 stations are used with 3 fixedC
C stations, but all 31 frequencies are used.  Fairshare is     C
C only 4 .  This small version has to integerize the ssource toC
C station arcs and the station to slack arc as picured in two- C
C stage network in chapter six.                                C
C MOST RECENT CHANGE:  24 Jan 91                               C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
      PARAMETER (NS=15,NF=31)
      INTEGER J,K,N
      REAL W1,W2,COEF,WCOEF1(NS,NF),WCOEF2(NF),HFDF,STAT,
     &FAIRSH

      OPEN (9,FILE='TEST.DAT',STATUS='UNKNOWN')
      OPEN (10,FILE='TOYWT.DAT',STATUS='OLD')
```

```
      OPEN (11,FILE='OBJT1.DAT',STATUS='OLD')

C     initialize variables
      HFDF = 120
      STAT = 10
      FAIRSH = 4

      DO 100 N=1,10
C       weightings for objective functions 1, and 2
        READ(10,700) W1,W2
C        WRITE(9,*) N,W1,W2

C       weighted coefficients for objective function 1
        DO 20 J=1,NS
          DO 30 K=1,NF
            READ(11,705) COEF
C           WRITE(9,*)J,K,COEF
            WCOEF1(J,K) = W1*COEF*1.0
C            WRITE(9,*)J,K,W1,W2,WCOEF1(J,K)
   30     CONTINUE
   20   CONTINUE

        REWIND(11)
C       weighted coefficients for excess coverage of freq k
        DO 40 K=1,NF
            WCOEF2(K) = W2 * (-1)
C            WRITE(9,*) W2,WCOEF2(K)
   40   CONTINUE

C     open file to write sas input for each weighting N
```

```
 IF (N.EQ.1) OPEN (12,FILE='[JSTEPPE.THESIS.saslp]sL1W1n.SAS',
&                      STATUS='NEW')
 IF (N.EQ.2) OPEN (12,FILE='[JSTEPPE.THESIS.SASLP]sL1W2n.SAS',
&                      STATUS='NEW')
 IF (N.EQ.3) OPEN (12,FILE='[JSTEPPE.THESIS.SASLP]sL1W3n.SAS',
&                      STATUS='NEW')
 IF (N.EQ.4) OPEN (12,FILE='[JSTEPPE.THESIS.SASLP]sL1W4n.SAS',
&                      STATUS='NEW')
 IF (N.EQ.5) OPEN (12,FILE='[JSTEPPE.THESIS.SASLP]sL1W5n.SAS',
&                      STATUS='NEW')
 IF (N.EQ.6) OPEN (12,FILE='[JSTEPPE.THESIS.SASLP]sL1W6n.SAS',
&                      STATUS='NEW')
 IF (N.EQ.7) OPEN (12,FILE='[JSTEPPE.THESIS.SASLP]sL1W7n.SAS',
&                      STATUS='NEW')
 IF (N.EQ.8) OPEN (12,FILE='[JSTEPPE.THESIS.SASLP]sL1W8n.SAS',
&                      STATUS='NEW')
 IF (N.EQ.9) OPEN (12,FILE='[JSTEPPE.THESIS.SASLP]sL1W9n.SAS',
&                      STATUS='NEW')
 IF (N.EQ.10) OPEN(12,FILE='[JSTEPPE.THESIS.SASLP]sL1W10n.SAS',
&                      STATUS='NEW')
 IF (N.EQ.11) OPEN(12,FILE='[JSTEPPE.THESIS.SASLP]sL1W11n.SAS',
&                      STATUS='NEW')
 IF (N.EQ.12) OPEN(12,FILE='[JSTEPPE.THESIS.SASLP]sL1W12n.SAS',
&                      STATUS='NEW')
 IF (N.EQ.13) OPEN(12,FILE='[JSTEPPE.THESIS.SASLP]sL1W13n.SAS',
&                      STATUS='NEW')
 IF (N.EQ.14) OPEN(12,FILE='[JSTEPPE.THESIS.SASLP]sL1W14n.SAS',
&                      STATUS='NEW')
 IF (N.EQ.15) OPEN(12,FILE='[JSTEPPE.THESIS.SASLP]sL1W15n.SAS',
&                      STATUS='NEW')
```

```
      IF (N.EQ.16) OPEN(12,FILE='[JSTEPPE.THESIS.SASLP]sL1W16n.SAS',
     &                    STATUS='NEW')
      IF (N.EQ.17) OPEN(12,FILE='[JSTEPPE.THESIS.SASLP]sL1W17n.SAS',
     &                    STATUS='NEW')
      IF (N.EQ.18) OPEN(12,FILE='[JSTEPPE.THESIS.SASLP]sL1W18n.SAS',
     &                    STATUS='NEW')
      IF (N.EQ.19) OPEN(12,FILE='[JSTEPPE.THESIS.SASLP]sL1W19n.SAS',
     &                    STATUS='NEW')
      IF (N.EQ.20) OPEN(12,FILE='[JSTEPPE.THESIS.SASLP]sL1W20n.SAS',
     &                    STATUS='NEW')
      IF (N.EQ.21) OPEN(12,FILE='[JSTEPPE.THESIS.SASLP]sL1W21n.SAS',
     &                    STATUS='NEW')

C     write the SAS input file

      WRITE (12,*) 'OPTIONS LINESIZE=78;'
      WRITE (12,*)
      WRITE (12,*) 'TITLE ''LOCATING A SAR NETWORK WITH GOOD HFDF
     & FREQUENCY ASSIGNMENTS'';'
      WRITE (12,*)
      WRITE (12,*) 'DATA SAR;'
      WRITE (12,*)
      WRITE (12,*) 'INPUT _TYPE_ $ _COL_ $ _ROW_ $
     & _COEF_;'
      WRITE (12,*)
      WRITE (12,*) 'CARDS;'
C     create the objective function

C     objective function one data and upperbounds for obj vars.
      WRITE(12,*) 'MAX           .       PROFIT       .'
```

```
      DO 110 J=1,NS
        DO 115 K=1,NF
         IF ((J .LT. 10) .AND. (K .LT.10)) THEN
          WRITE (12,750) '.          S',J,'F',K,'     PROFIT '
     &    ,WCOEF1(J,K)
         ELSEIF (J .LT. 10) THEN
          WRITE (12,751) '.          S',J,'F',K,'     PROFIT '
     &    ,WCOEF1(J,K)
         ELSEIF (K .LT. 10) THEN
          WRITE (12,752) '.          S',J,'F',K,'     PROFIT '
     &    ,WCOEF1(J,K)
         ELSE
          WRITE (12,753) '.          S',J,'F',K,'     PROFIT '
     &    ,WCOEF1(J,K)
         ENDIF
  115   CONTINUE
  110 CONTINUE

C     objective function two data & upperbounds for obj vars
      DO 125 K=1,NF
        IF (K .LT. 10 ) THEN
          WRITE (12,760) '.          F',K,'EX      PROFIT',WCOEF2(K)
          WRITE (12,765) 'UPPERBD    F',K,'EX      UPF',K
     &    ,'EX      6.0'
        ELSE
          WRITE (12,761) '.          F',K,'EX      PROFIT',WCOEF2(K)
          WRITE (12,766) 'UPPERBD    F',K,'EX     UPF',K
     &    ,'EX      6.0'
        ENDIF
  125 CONTINUE
```

```
C     source to receiving station constraint
      WRITE(12,*) 'EQ                .          SOST      .'
      WRITE(12,*) '.           _RHS_          SOST      ',STAT
      DO 130 J=1,NS
        IF (J .LT. 10) THEN
          WRITE(12,770) '.          SOST',J,'     SOST          1.0'
        ELSE
          WRITE(12,771) '.          SOST',J,'     SOST          1.0'
        ENDIF
  130 CONTINUE

C     conservation of flow at the receiving stations
      DO 135 J=1,NS
C  5 true fixed stations are 1,8,15,27,28 as defined by thesis prob
        IF ((J .EQ. 1) .OR. (J .EQ. 8) .OR. (J .EQ. 15) .OR.
     &  (J .EQ.27) .OR. (J .EQ. 28)) THEN

          IF (J .LT. 10) THEN
            WRITE(12,775)'EQ                .        CONST',J
     &      ,'     .  '
            WRITE(12,775)'.            _RHS_        CONST',J
     &      ,'      1.0'
            WRITE(12,780)'.            SOST',J,'      CONST',J
     &      ,'      3.0'
            WRITE(12,785)'UPPERBD      SOST',J,'     UPSOST',J
     &      ,'      1.0'
            WRITE(12,790)'.            ST',J,'SL      CONST',J
     &      ,'     -1.0'
            WRITE(12,730)'BINARY       ST',J,'SL    BINARY      1'
```

```
      WRITE(12,800)'.          ST',J,'S',J,'      CONST',J
&     ,'    -1.0'
      WRITE(12,800)'UPPERBD    ST',J,'S',J,'     UPPERS',J
&     ,'     2.0'
     ELSE
      WRITE(12,776)'EQ                .       CONST',J
&     ,'     . '
      WRITE(12,776)'.          _RHS_       CONST',J
&     ,'     1.0'
      WRITE(12,781)'.         SGST',J,'     CONST',J
&     ,'     3.0'
      WRITE(12,786)'UPPERBD   SOST',J,'    UPSOST',J
&     ,'     1.0'
      WRITE(12,791)'.         ST',J,'SL     CONST',J
&     ,'    -1.0'
      WRITE(12,731)'BINARY    ST',J,'SL    BINARY      1'
      WRITE(12,803)'.        ST',J,'S',J,'     CONST',J
&     ,'    -1.0'
      WRITE(12,803)'UPPERBD  ST',J,'S',J,'    UPPERS',J
&     ,'     2.0'
     ENDIF

   ELSE

    IF (J .LT. 10) THEN
      WRITE(12,775)'EQ                .       CONST',J
&     ,'      .'
      WRITE(12,775)'.          _RHS_       CONST',J
&     ,'     0.0'
      WRITE(12,780)'.          SOST',J,'      CONST',J
```

```
&      ,'    2.0'
        WRITE(12,730)'BINARY  SOST',J,'     BINARY      1'
        WRITE(12,790)'.         ST',J,'SL   . CONST',J
&      ,'   -1.0'
        WRITE(12,730)'BINARY    ST',J,'SL   BINARY      1'
        WRITE (12,800)'.        ST',J,'S',J,'     CONST',J
&      ,'   -1.0'
        WRITE (12,800)'UPPERBD   ST',J,'S',J,'    UPPERS',J
&      ,'    2.0'
      ELSE
        WRITE(12,776)'EQ              .      CONST',J
&      ,'      .'
        WRITE(12,776)'.          _RHS_      CONST',J
&      ,'    0.0'
        WRITE(12,781)'.         SOST',J,'    CONST',J
&      ,'    2.0'
        WRITE(12,731)'BINARY  SOST',J,'     BINARY      1'
        WRITE(12,791)'.         ST',J,'SL    CONST',J
&      ,'   -1.0'
        WRITE(12,731)'BINARY    ST',J,'SL   BINARY      1'
        WRITE(12,803)'.         ST',J,'S',J,'    CONST',J
&      ,'   -1.0'
        WRITE(12,803)'UPPERBD  ST',J,'S',J,'   UPPERS',J
&      ,'    2.0'
      ENDIF
     ENDIF
 135 CONTINUE

C     conservation of flow at slack
      WRITE(12,*) 'EQ              .      CONSL    .'
```

```
      WRITE(12,*) '.          _RHS_     CONSL    5.0'
      DO 175 J=1,NS
        IF (J .LT. 10) THEN
          WRITE(12,830) '.          ST',J,'SL      CONSL     1.0'
        ELSE
          WRITE(12,831) '.          ST',J,'SL      CONSL     1.0'
        ENDIF
  175 CONTINUE
      WRITE(12,*) '.          SLSI      CONSL     -1.0'
      WRITE(12,*) 'UPPERBD    SLSI      UPSLSI    20.0'

C     conservation of flow at restationing node
      DO 178 J=1,NS
        IF (J .LT. 10) THEN
          WRITE(12,775)'EQ          .         CONS',J,'     .  '
          WRITE(12,775)'.          _RHS_      CONS',J,'    0.0 '
          WRITE(12,800)'.          ST',J,'S',J,'      CONS',J
     &    ,'     8.0'
          DO 176 K=1,NF
            IF (K .LT. 10) THEN
              WRITE(12,850)'.          S',J,'F',K,'       CONS'
     &        ,J,'    -1.0'
            ELSE
              WRITE(12,851)'.          S',J,'F',K,'       CONS'
     &        ,J,'    -1.0'
            ENDIF
  176     CONTINUE
        ELSE
          WRITE(12,776)'EQ          .         CONS',J,'     . '
          WRITE(12,776)'.          _RHS_      CONS',J,'    0.0'
```

```
        WRITE(12,803)'.        ST',J,'S',J,'      CONS',J,
     &  '     8.0'
        DO 177 K=1,NF
          IF (K .LT. 10) THEN
            WRITE(12,852)'.         S',J,'F',K,'       CONS'
     &         ,J,'     -1.0'
          ELSE
            WRITE(12,853)'.        S',J,'F',K,'       CONS'
     &         ,J,'     -1.0'
          ENDIF
 177    CONTINUE
      ENDIF
 178 CONTINUE

C     conservation of flow at each freouency
      DO 180 K=1,NF
        IF (K .LT. 10) THEN
          WRITE(12,835) 'EQ             .       CONF',K,'     .'
          WRITE(12,835) '.          _RHS_       CONF',K,'   0.0'
          WRITE(12,840) '.            F',K,'EX       CONF',K,'     -1.0'
          WRITE(12,840) '.            F',K,'NE       CONF',K,'     -1.0'
          WRITE(12,845) 'UPPERBD      F',K,'NE     UPF',K,'NE     4.0'
        ELSE
          WRITE(12,836) 'EQ             .      CONF',K,'     .'
          WRITE(12,836) '.          _RHS_      CONF',K,'   0.0'
          WRITE(12,841) '.           F',K,'EX      CONF',K,'     -1.0'
          WRITE(12,841) '.           F',K,'NE      CONF',K,'     -1.0'
          WRITE(12,846) 'UPPERBD     F',K,'NE     UPF',K,'NE     4.0'
        ENDIF
        DO 185 J=1,NS
```

```
      IF ((J .LT. 10) .AND. (K .LT. 10)) THEN
        WRITE(12,850) '.          S',J,'F',K,'       CONF',K
     &      ,'      1.0'
        WRITE(12,720)'UPPERBD    S',J,'F',K,'     UPST',J,
     &      'F',K,'     1.0'
      ELSE IF (J .LT. 10) THEN
        WRITE(12,857) '.          S',J,'F',K,'      CONF',K
     &      ,'      1.0'
        WRITE(12,721)'UPPERBD   S',J,'F',K,'     UPST',J,
     &      'F',K,'     1.0'
      ELSE IF (K .LT. 10) THEN
        WRITE(12,858) '.          S',J,'F',K,'       CONF',K
     &      ,'      1.0'
        WRITE(12,722)'UPPERBD   S',J,'F',K,'     UPST',J,
     &      'F',K,'     1.0'
      ELSE
        WRITE(12,853) '.         S',J,'F',K,'       CONF',K
     &      ,'      1.0'
        WRITE(12,723)'UPPERBD  S',J,'F',K,'    UPST',J,
     &      'F',K,'     1.0'
      ENDIF
 185  CONTINUE
 180 CONTINUE

C     conservation of flow at excess coverage
      WRITE(12,*) 'EQ              .        CONEX     .'
      WRITE(12,*) '.          _RHS_       CONEX   0.0'
      DO 195 K=1,NF
        IF (K .LT. 10) THEN
          WRITE(12,855) '.          F',K,'EX       CONEX     1.0'
```

```
          ELSE
            WRITE(12,856) '.          F',K,'EX       CONEX     1.0'
          ENDIF
  195 CONTINUE
        WRITE(12,*) '.          EXSI        CONEX     -1.0'
        WRITE(12,*) 'UPPERBD    EXSI       UPEXSI    80'

C       conservation of flow at nonexcess coverage
        WRITE(12,*) 'EQ              .      CONNE      .'
        WRITE(12,*) '.          _RHS_       CONNE     0.0'
        DO 200 K=1,NF
          IF (K .LT. 10) THEN
            WRITE(12,855) '.          F',K,'NE       CONNE     1.0'
          ELSE
            WRITE(12,856) '.          F',K,'NE       CONNE     1.0'
          ENDIF
  200 CONTINUE
        WRITE(12,*) '.          NESI        CONNE     -1.0'
        WRITE(12,*) 'UPPERBD    NESI       UPNESI    120.0'

C       sink constraint
        WRITE(12,*) 'EQ              .       SINK      .'
        WRITE(12,*) '.          _RHS_        SINK     120.0'
        WRITE(12,*) '.          NESI         SINK     1.0'
        WRITE(12,*) '.          EXSI         SINK     1.0'
        WRITE(12,*) '.          SLSI         SINK     1.0'


        WRITE (12,*) '; '
        WRITE (12,*) 'PROC LP SPARSEDATA MAXIT1=10000 MAXIT2=100000 IMAXIT
```

```
&=99999999'
  WRITE (12,*) 'PRINTFREQ=200;'

  WRITE (12,*) 'RUN;'

  CLOSE (12)

100 CONTINUE

700 FORMAT (F7.5,F7.5)
705 FORMAT (10X,F14.7)
720 FORMAT (1X,A13,I1,A1,I1,A10,I1,A1,I1,A8)
721 FORMAT (1X,A12,I1,A1,I2,A9,I1,A1,I2,A8)
722 FORMAT (1X,A12,I2,A1,I1,A9,I2,A1,I1,A8)
723 FORMAT (1X,A11,I2,A1,I2,A8,I2,A1,I2,A8)
730 FORMAT (1X,A13,I1,A19)
731 FORMAT (1X,A12,I2,A19)
740 FORMAT (1X,A15,I1,A20)
741 FORMAT (1X,A14,I2,A20)
750 FORMAT (1X,A13,I1,A1,I1,A13,F10.6)
751 FORMAT (1X,A12,I1,A1,I2,A13,F10.6)
752 FORMAT (1X,A12,I2,A1,I1,A13,F10.6)
753 FORMAT (1X,A11,I2,A1,I2,A13,F10.6)
755 FORMAT (1X,A13,I1,A1,I1,A8,I1,A1,I1,A8)
756 FORMAT (1X,A12,I1,A1,I2,A7,I1,A1,I2,A8)
757 FORMAT (1X,A12,I2,A1,I1,A7,I2,A1,I1,A8)
758 FORMAT (1X,A11,I2,A1,I2,A6,I2,A1,I2,A8)
760 FORMAT (1X,A13,I1,A14,F10.6)
761 FORMAT (1X,A12,I2,A14,F10.6)
765 FORMAT (1X,A13,I1,A11,I1,A11)
```

```
766 FORMAT (1X,A12,I2,A10,I2,A11)
770 FORMAT (1X,A15,I1,A21)
771 FORMAT (1X,A14,I2,A21)
773 FORMAT (1X,A15,I1,A12,I1,A8)
774 FORMAT (1X,A14,I2,A11,I2,A8)
775 FORMAT (1X,A28,I1,A8)
776 FORMAT (1X,A27,I2,A8)
780 FORMAT (1X,A15,I1,A11,I1,A8)
781 FORMAT (1X,A14,I2,A10,I2,A8)
785 FORMAT (1X,A15,I1,A11,I1,A8)
786 FORMAT (1X,A14,I2,A10,I2,A8)
790 FORMAT (1X,A13,I1,A13,I1,A8)
791 FORMAT (1X,A12,I2,A12,I2,A8)
795 FORMAT (1X,A13,I1,A11,I1,A9)
796 FORMAT (1X,A12,I2,A10,I2,A9)
800 FORMAT (1X,A13,I1,A1,I1,A11,I1,A8)
803 FORMAT (1X,A11,I2,A1,I2,A10,I2,A8)
830 FORMAT (1X,A13,I1,A22)
831 FORMAT (1X,A12,I2,A22)
835 FORMAT (1X,A27,I1,A6)
836 FORMAT (1X,A26,I2,A6)
840 FORMAT (1X,A13,I1,A13,I1,A9)
841 FORMAT (1X,A12,I2,A12,I2,A9)
845 FORMAT (1X,A13,I1,A11,I1,A10)
846 FORMAT (1X,A12,I2,A10,I2,A10)
850 FORMAT (1X,A13,I1,A1,I1,A12,I1,A8)
851 FORMAT (1X,A12,I1,A1,I2,A12,I1,A8)
852 FORMAT (1X,A12,I2,A1,I1,A12,I2,A8)
853 FORMAT (1X,A11,I2,A1,I2,A11,I2,A8)
855 FORMAT (1X,A13,I1,A22)
```

```
856 FORMAT (1X,A12,I2,A22)
857 FORMAT (1X,A12,I1,A1,I2,A11,I2,A8)
858 FORMAT (1X,A12,I2,A1,I1,A12,I1,A8)
    END
```

Appendix D. *Stage–One SAS LP Input File*

SAS LP requires an input file. This FORTRAN program generates an input file in the sparse format for the stage one of the two-stage formulation. A copy of this code called *append4* is provided on floppy disk 1 in the FORTRAN directory.

```
      PROGRAM stage1

CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C              THESIS STAGE ONE OPTIMIZATION                C
C PURPOSE: THIS PROGRAM WRITES THE INPUT FILE FOR SAS PROC LP C
C THE PROBLEM IS MULTIOBJECTIVE AND THEREFORE SEVERAL WEIGHT- C
C INGS OF THE OBJECTIVE FUNCTION WILL BE INVESTIGATED.        C
C TOTWT.DAT      HAS THE OBJECTIVE FUNCTION WEIGHTINGS        C
C OBJT*.DAT      HAS THE OBJECTIVE FUNCTION COEFFICIENTS FOR  C
C                TIME PERIOD '*' FOR OBJECTIVE ONE, WITH NO   C
C                WEIGHTINGS APPLIED.                          C
C M*W&.SAS       IS THE SAS INPUT FILE FOR TIME PERIOD '*'    C
C                AND WEIGHT '&'                               C
C MULTICOMMODITY FLOW IDEA USED FOR BUNDLING.  FIRST SOURCE   C
C GIVES BUNDLE TO 20 STATIONS, 2ND SOURCE ALLOCATES 10 ADDITNL C
C BUNDLES TO STATIONS THAT ALREADY HAVE ONE BUNDLE FROM SOURCE C
C ONE.                                                        C
C This program generates matrix input for the stage one optimi C
C zation. The twenty stations selected by stage one will be   C
C located.  Currently there are 25 binary variables (SOSTj).  C
C Five stations are already fixed: 1, 8, 15, 27, and 28.      C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC

      PARAMETER (NS=30,NF=31)
      INTEGER J,K,N
      REAL W1,W2,COEF,WCOEF1(NS,NF),WCOEF2(NF),HFDF,STAT,
     &FAIRSH

      OPEN (9,FILE='TEST.DAT',STATUS='UNKNOWN')
```

```
      OPEN (10,FILE='TOYWT.DAT',STATUS='OLD')
      OPEN (11,FILE='OBJT7.DAT',STATUS='OLD')

C     initialize variables
C     stat1 represents that the 20 stations chosen will each get
C     one bundle of HFDFs

      HFDF = 240
      STAT1 = 20
      FAIRSH = 8

      DO 100 N=1,24
C       weightings for objective functions 1, and 2
        READ(10,700) W1,W2
C        WRITE(9,*) N,W1,W2

C       weighted coefficients for objective function 1
        DO 20 J=1,NS
          DO 30 K=1,NF
            READ(11,705) COEF
C           WRITE(9,*)J,K,COEF
            WCOEF1(J,K) = W1*COEF*1.0
C            WRITE(9,*)J,K,W1,W2,WCOEF1(J,K)
 30       CONTINUE
 20     CONTINUE

        REWIND(11)
C       weighted coefficients for excess coverage of freq k
        DO 40 K=1,NF
            WCOEF2(K) = W2 * (-1)
```

```
C             WRITE(9,*) W2,WCOEF2(K)
  40    CONTINUE

C      open file to write sas input for each weighting N
       IF (N.EQ.1) OPEN (12,FILE='[JSTEPPE.THESIS.SASLP]M7W1.SAS',
     &                      STATUS='NEW')
       IF (N.EQ.2) OPEN (12,FILE='[JSTEPPE.THESIS.SASLP]M7W2.SAS',
     &                      STATUS='NEW')
       IF (N.EQ.3) OPEN (12,FILE='[JSTEPPE.THESIS.SASLP]M7W3.SAS',
     &                      STATUS='NEW')
       IF (N.EQ.4) OPEN (12,FILE='[JSTEPPE.THESIS.SASLP]M7W4.SAS',
     &                      STATUS='NEW')
       IF (N.EQ.5) OPEN (12,FILE='[JSTEPPE.THESIS.SASLP]M7W5.SAS',
     &                      STATUS='NEW')
       IF (N.EQ.6) OPEN (12,FILE='[JSTEPPE.THESIS.SASLP]M7W6.SAS',
     &                      STATUS='NEW')
       IF (N.EQ.7) OPEN (12,FILE='[JSTEPPE.THESIS.SASLP]M7W7.SAS',
     &                      STATUS='NEW')
       IF (N.EQ.8) OPEN (12,FILE='[JSTEPPE.THESIS.SASLP]M7W8.SAS',
     &                      STATUS='NEW')
       IF (N.EQ.9) OPEN (12,FILE='[JSTEPPE.THESIS.SASLP]M7W9.SAS',
     &                      STATUS='NEW')
       IF (N.EQ.10) OPEN(12,FILE='[JSTEPPE.THESIS.SASLP]M7W10.SAS',
     &                      STATUS='NEW')
       IF (N.EQ.11) OPEN(12,FILE='[JSTEPPE.THESIS.SASLP]M7W11.SAS',
     &                      STATUS='NEW')
       IF (N.EQ.12) OPEN(12,FILE='[JSTEPPE.THESIS.SASLP]M7W12.SAS',
     &                      STATUS='NEW')
       IF (N.EQ.13) OPEN(12,FILE='[JSTEPPE.THESIS.SASLP]M7W13.SAS',
     &                      STATUS='NEW')
```

```
      IF (N.EQ.14) OPEN(12,FILE='[JSTEPPE.THESIS.SASLP]M7W14.SAS',
     &                  STATUS='NEW')
      IF (N.EQ.15) OPEN(12,FILE='[JSTEPPE.THESIS.SASLP]M7W15.SAS',
     &                  STATUS='NEW')
      IF (N.EQ.16) OPEN(12,FILE='[JSTEPPE.THESIS.SASLP]M7W16.SAS',
     &                  STATUS='NEW')
      IF (N.EQ.17) OPEN(12,FILE='[JSTEPPE.THESIS.SASLP]M7W17.SAS',
     &                  STATUS='NEW')
      IF (N.EQ.18) OPEN(12,FILE='[JSTEPPE.THESIS.SASLP]M7W18.SAS',
     &                  STATUS='NEW')
      IF (N.EQ.19) OPEN(12,FILE='[JSTEPPE.THESIS.SASLP]M7W19.SAS',
     &                  STATUS='NEW')
      IF (N.EQ.20) OPEN(12,FILE='[JSTEPPE.THESIS.SASLP]M7W20.SAS',
     &                  STATUS='NEW')
      IF (N.EQ.21) OPEN(12,FILE='[JSTEPPE.THESIS.SASLP]M7W21.SAS',
     &                  STATUS='NEW')
      IF (N.EQ.22) OPEN(12,FILE='[JSTEPPE.THESIS.SASLP]M7W22.SAS',
     &                  STATUS='NEW')
      IF (N.EQ.23) OPEN(12,FILE='[JSTEPPE.THESIS.SASLP]M7W23.SAS',
     &                  STATUS='NEW')
      IF (N.EQ.24) OPEN(12,FILE='[JSTEPPE.THESIS.SASLP]M7W24.SAS',
     &                  STATUS='NEW')
C    write the SAS input file

      WRITE (12,*) 'OPTIONS LINESIZE=78;'
      WRITE (12,*)
      WRITE (12,*) 'TITLE ''LOCATING A SAR NETWORK WITH GOOD HFDF
     & FREQUENCY ASSIGNMENTS'';'
      WRITE (12,*)
      WRITE (12,*) 'DATA SAR;'
```

```
      WRITE (12,*)
      WRITE (12,*) 'INPUT _TYPE_ $ _COL_ $ _ROW_ $
     & _COEF_;'
      WRITE (12,*)
      WRITE (12,*) 'CARDS;'
C     create the objective function

C     objective function one data & upperbounds for obj vars.
      WRITE (12,*) 'MAX                    .       PROFIT      .'
      DO 110 J=1,NS
        DO 115 K=1,NF
          IF ((J .LT. 10) .AND. (K .LT.10)) THEN
            WRITE (12,750) '.          ST',J,'F',K,'      PROFIT '
     &      ,WCOEF1(J,K)
            WRITE (12,755) 'UPPERBD    ST',J,'F',K,'    UPST',J,'F',K
     &      ,'     1.0'
          ELSEIF (J .LT. 10) THEN
            WRITE (12,751) '.         ST',J,'F',K,'      PROFIT '
     &      ,WCOEF1(J,K)
            WRITE (12,756) 'UPPERBD   ST',J,'F',K,'    UPST',J,'F',K
     &      ,'     1.0'
          ELSEIF (K .LT. 10) THEN
            WRITE (12,752) '.         ST',J,'F',K,'      PROFIT '
     &      ,WCOEF1(J,K)
            WRITE (12,757) 'UPPERBD   ST',J,'F',K,'    UPST',J,'F',K
     &      ,'     1.0'
          ELSE
            WRITE (12,753) '.        ST',J,'F',K,'      PROFIT '
     &      ,WCOEF1(J,K)
            WRITE (12,758) 'UPPERBD  ST',J,'F',K,'   UPST',J,'F',K
```

```
     &     ,'     1.0'

        ENDIF
 115    CONTINUE
 110 CONTINUE

C     objective function two data & upperbounds for obj vars
      DO 125 K=1,NF
        IF (K .LT. 10 ) THEN
          WRITE (12,760) '.          F',K,'EX     PROFIT',WCOEF2(K)
          WRITE (12,765) 'UPPERBD    F',K,'EX      UPF',K
     &    ,'EX     16.0'
        ELSE
          WRITE (12,761) '.         F',K,'EX     PROFIT',WCOEF2(K)
          WRITE (12,766) 'UPPERBD   F',K,'EX     UPF',K
     &    ,'EX     16.0'
        ENDIF
 125 CONTINUE

C     source ONE to receiving station constraint
C     20 stations will get one bundle of HFDF resources
      WRITE(12,*) 'EQ              .       S01ST      .'
      WRITE(12,*) '.         _RHS_      S01ST      ',STAT1
      DO 130 J=1,NS
        IF (J .LT. 10) THEN
          WRITE(12,770) '.        S01ST',J,'    S01ST         1.0'
        ELSE
          WRITE(12,771) '.       S01ST',J,'    S01ST         1.0'
        ENDIF
 130 CONTINUE
```

```
C     conservation of flow at the receiving stations
      DO 135 J=1,NS
C     fixed stations  defined for thesis problem
        IF ((J .EQ. 1) .OR. (J .EQ. 8) .OR. (J .EQ. 15) .OR.
     & (J .EQ.27) .OR. (J .EQ. 28)) THEN

          IF (J .LT. 10) THEN
            WRITE(12,775)'EQ                .       CONST',J
     &      ,'     . '
            WRITE(12,775)'.           _RHS_       CONST',J
     &      ,'     1.0'
            WRITE(12,780)'.          S01ST',J,'      CONST',J
     &      ,'     9.0'
            WRITE(12,785)'UPPERBD   S01ST',J,'    UPS01ST',J
     &      ,'     1.0'
            DO 140 K=1,NF
              IF (K .LT. 10) THEN
                WRITE (12,800)'.          ST',J,'F',K,'     CONST',J
     &          ,'    -1.0'
              ELSE
                WRITE(12,802)'.          ST',J,'F',K,'     CONST',J
     &          ,'    -1.0'
              ENDIF
 140        CONTINUE
          ELSE
            WRITE(12,776)'EQ                .       CONST',J
     &      ,'     . '
            WRITE(12,776)'.           _RHS_       CONST',J
     &      ,'     1.0'
```

```
        WRITE(12,781)'.        S01ST',J,'     CONST',J
     &  ,'     9.0'
        WRITE(12,786)'UPPERBD  S01ST',J,'    UPSOST',J
     &  ,'     1.0'
        DO 145 K=1,NF
          IF (K .LT. 10) THEN
            WRITE(12,801)'.        ST',J,'F',K,'     CONST',J
     &      ,'    -1.0'
          ELSE
            WRITE(12,803)'.        ST',J,'F',K,'     CONST',J
     &      ,'    -1.0'
          ENDIF
145     CONTINUE
       ENDIF

      ELSE

       IF (J .LT. 10) THEN
        WRITE(12,775)'EQ              .       CONST',J
     &  ,'        .'
        WRITE(12,775)'.          _RHS_       CONST',J
     &  ,'     0.0'
        WRITE(12,780)'.        S01ST',J,'     CONST',J
     &  ,'     8.0'
        WRITE(12,740)'BINARY    S01ST',J,' BINARY    1'
        DO 150 K=1,NF
          IF (K .LT. 10) THEN
            WRITE (12,800)'.        ST',J,'F',K,'     CONST',J
     &      ,'    -1.0'
          ELSE
```

```
              WRITE(12,802)'.        ST',J,'F',K,'    CONST',J
     &        ,'    -1.0'
            ENDIF
 150      CONTINUE
          ELSE
          WRITE(12,776)'EQ                .      CONST',J
     &    ,'       .'
          WRITE(12,776)'.          _RHS_      CONST',J
     &    ,'    0.0'
          WRITE(12,781)'.        S01ST',J,'    CONST',J
     &    ,'    8.0'
          WRITE(12,741)'BINARY   S01ST',J,'  BINARY   1'
          DO 155 K=1,NF
            IF (K .LT. 10) THEN
              WRITE(12,801)'.        ST',J,'F',K,'    CONST',J
     &        ,'    -1.0'
            ELSE
              WRITE(12,803)'.        ST',J,'F',K,'    CONST',J
     &        ,'    -1.0'
            ENDIF
 155      CONTINUE
        ENDIF
      ENDIF
 135 CONTINUE

C     conservation of flow at each frequency
      DO 180 K=1,NF
        IF (K .LT. 10) THEN
          WRITE(12,835) 'EQ              .      CONF',K,'     .'
          WRITE(12,835) '.         _RHS_      CONF',K,'   0.0'
```

```
      WRITE(12,840) '.          F',K,'EX       CONF',K,'    -1.0'
      WRITE(12,840) '.          F',K,'NE       CONF',K,'    -1.0'
      WRITE(12,845) 'UPPERBD    F',K,'NE      UPF',K,'NE    8.0'
    ELSE
      WRITE(12,836) 'EQ                 .     CONF',K,'     .'
      WRITE(12,836) '.         _RHS_     CONF',K,'   0.0'
      WRITE(12,841) '.         F',K,'EX     CONF',K,'    -1.0'
      WRITE(12,841) '.         F',K,'NE     CONF',K,'    -1.0'
      WRITE(12,846) 'UPPERBD   F',K,'NE    UPF',K,'NE    8.0'
    ENDIF
    DO 185 J=1,NS
      IF ((J .LT. 10) .AND. (K .LT. 10)) THEN
        WRITE(12,850) '.         ST',J,'F',K,'        CONF',K
 &      ,'     1.0'
      ELSE IF (J .LT. 10) THEN
        WRITE(12,851) '.        ST',J,'F',K,'       CONF',K
 &      ,'     1.0'
      ELSE IF (K .LT. 10) THEN
        WRITE(12,852) '.        ST',J,'F',K,'        CONF',K
 &      ,'     1.0'
      ELSE
        WRITE(12,853) '.       ST',J,'F',K,'       CONF',K
 &      ,'     1.0'
      ENDIF
185   CONTINUE
180 CONTINUE

C     conservation of flow at excess coverage
      WRITE(12,*) 'EQ              .         CONEX     .'
      WRITE(12,*) '.         _RHS_       CONEX   0.0'
```

```
      DO 195 K=1,NF
        IF (K .LT. 10) THEN
          WRITE(12,855) '.           F',K,'EX       CONEX     1.0'
        ELSE
          WRITE(12,856) '.          F',K,'EX       CONEX     1.0'
        ENDIF
  195 CONTINUE
      WRITE(12,*) '.           EXSI       CONEX     -1.0'
      WRITE(12,*) 'UPPERBD     EXSI      UPEXSI    160.0'

C     conservation of flow at nonexcess coverage
      WRITE(12,*) 'EQ              .       CONNE      .'
      WRITE(12,*) '.          _RHS_       CONNE     0.0'
      DO 200 K=1,NF
        IF (K .LT. 10) THEN
          WRITE(12,855) '.           F',K,'NE       CONNE     1.0'
        ELSE
          WRITE(12,856) '.          F',K,'NE       CONNE     1.0'
        ENDIF
  200 CONTINUE
      WRITE(12,*) '.           NESI       CONNE     -1.0'
      WRITE(12,*) 'UPPERBD     NESI      UPNESI    160.0'

C     sink constraint
      WRITE(12,*) 'EQ              .        SINK      .'
      WRITE(12,*) '.          _RHS_        SINK     160.0'
      WRITE(12,*) '.           NESI        SINK     1.0'
      WRITE(12,*) '.           EXSI        SINK     1.0'
```

```
      WRITE (12,*) '; '
      WRITE (12,*) 'PROC LP SPARSEDATA POUT=SOLUTION MAXIT1=10000
     C MAXIT2=999999 IMAXIT=99999999'
      WRITE (12,*) 'PRINTFREQ=500;'
      WRITE (12,*) 'RUN;'

      CLOSE (12)

  100 CONTINUE

  700 FORMAT (2(F7.4))
  705 FORMAT (10X,F14.7)
  730 FORMAT (1X,A28,I1,A8)
  731 FORMAT (1X,A27,I2,A8)
  735 FORMAT (1X,A15,I1,A8,I1,A6)
  736 FORMAT (1X,A15,I2,A8,I2,A6)
  740 FORMAT (1X,A15,I1,A13)
  741 FORMAT (1X,A14,I2,A13)
  750 FORMAT (1X,A13,I1,A1,I1,A13,F10.6)
  751 FORMAT (1X,A12,I1,A1,I2,A13,F10.6)
  752 FORMAT (1X,A12,I2,A1,I1,A13,F10.6)
  753 FORMAT (1X,A11,I2,A1,I2,A13,F10.6)
  755 FORMAT (1X,A13,I1,A1,I1,A8,I1,A1,I1,A8)
  756 FORMAT (1X,A12,I1,A1,I2,A7,I1,A1,I2,A8)
  757 FORMAT (1X,A12,I2,A1,I1,A7,I2,A1,I1,A8)
  758 FORMAT (1X,A11,I2,A1,I2,A6,I2,A1,I2,A8)
  760 FORMAT (1X,A13,I1,A13,F10.6)
  761 FORMAT (1X,A12,I2,A13,F10.6)
  765 FORMAT (1X,A13,I1,A11,I1,A11)
  766 FORMAT (1X,A12,I2.A10,I2,A11)
```

```
770 FORMAT (1X,A15,I1,A21)
771 FORMAT (1X,A14,I2,A21)
773 FORMAT (1X,A15,I1,A12,I1,A8)
774 FORMAT (1X,A14,I2,A11,I2,A8)
775 FORMAT (1X,A28,I1,A8)
776 FORMAT (1X,A27,I2,A8)
780 FORMAT (1X,A15,I1,A11,I1,A8)
781 FORMAT (1X,A14,I2,A10,I2,A8)
785 FORMAT (1X,A15,I1,A11,I1,A8)
786 FORMAT (1X,A14,I2,A10,I2,A8)
790 FORMAT (1X,A13,I1,A13,I1,A8)
791 FORMAT (1X,A12,I2,A12,I2,A8)
795 FORMAT (1X,A13,I1,A11,I1,A10)
796 FORMAT (1X,A12,I2,A10,I2,A10)
800 FORMAT (1X,A13,I1,A1,I1,A11,I1,A8)
802 FORMAT (1X,A12,I1,A1,I2,A11,I1,A8)
801 FORMAT (1X,A12,I2,A1,I1,A10,I2,A8)
803 FORMAT (1X,A11,I2,A1,I2,A10,I2,A8)
810 FORMAT (1X,A26,I1,I1,A6)
811 FORMAT (1X,A25,I2,I1,A6)
815 FORMAT (1X,A12,I1,A1,I1,I1,A10,I1,I1,A8)
816 FORMAT (1X,A10,I2,A1,I2,I1,A9,I2,I1,A8)
820 FORMAT (1X,A12,I1,I1,A1,I1,A10,I1,I1,A9)
821 FORMAT (1X,A11,I1,I1,A1,I2,A10,I1,I1,A9)
822 FORMAT (1X,A11,I2,I1,A1,I1,A9,I2,I1,A9)
823 FORMAT (1X,A10,I2,I1,A1,I2,A9,I2,I1,A9)
825 FORMAT (1X,A12,I1,I1,A1,I1,A8,I1,I1,A1,I1,A8)
826 FORMAT (1X,A11,I1,I1,A1,I2,A7,I1,I1,A1,I2,A8)
827 FORMAT (1X,A11,I2,I1,A1,I1,A7,I2,I1,A1,I1,A8)
828 FORMAT (1X,A10,I2,I1,A1,I2,A6,I2,I1,A1,I2,A8)
```

```
      /
830 FORMAT (1X,A13,I1,A22)
831 FORMAT (1X,A12,I2,A22)
835 FORMAT (1X,A27,I1,A6)
836 FORMAT (1X,A26,I2,A6)
840 FORMAT (1X,A13,I1,A13,I1,A9)
841 FORMAT (1X,A12,I2,A12,I2,A9)
845 FORMAT (1X,A13,I1,A11,I1,A9)
846 FORMAT (1X,A12,I2,A10,I2,A9)
850 FORMAT (1X,A13,I1,A1,I1,A12,I1,A8)
851 FORMAT (1X,A12,I1,A1,I2,A11,I2,A8)
852 FORMAT (1X,A12,I2,A1,I1,A12,I1,A8)
853 FORMAT (1X,A11,I2,A1,I2,A11,I2,A8)
855 FORMAT (1X,A13,I1,A22)
856 FORMAT (1X,A12,I2,A22)
    END
```

Appendix E. *Stage–Two SAS LP Input File*

SAS LP requires an input file. This FORTRAN program generates an input file in the sparse format for the second stage of the two-stage formulation. A copy of this code called *append5* is provided on floppy disk 1 in the FORTRAN directory.

```
      PROGRAM STAGE2

CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C               THESIS STAGE-TWO OPTIMIZATION                  C
C PURPOSE: THIS PROGRAM WRITES THE INPUT FILE FOR SAS PROC LP   C
C THE PROBLEM IS MULTIOBJECTIVE AND THEREFORE SEVERAL WEIGHT-   C
C INGS OF THE OBJECTIVE FUNCTION WILL BE INVESTIGATED.          C
C TOYWT.DAT      HAS THE OBJECTIVE FUNCTION WEIGHTINGS          C
C OBJT*.DAT      HAS THE OBJECTIVE FUNCTION COEFFICIENTS FOR    C
C                TIME PERIOD '*' FOR OBJECTIVE ONE, WITH NO     C
C                WEIGHTINGS APPLIED.                            C
C F*W&.SAS       IS THE SAS INPUT FILE FOR TIME PERIOD '*'      C
C                AND WEIGHT '&'                                 C
C MULTICOMMODITY FLOW IDEA USED FOR BUNDLING. tWO STAGE         C
C PROCESS IS USED.  IN STAGE ONE 20 BUNDLES ARE ASSIGNED, ONE   C
C TO EACH OF 20 STATIONS, CHOSEN AMONG 30 STATIONS.  STAGE TWO C
C ASSIGNS 10 MORE BUNDLES, EITHER ONE OR ZERO TO EACH OF THE    C
C TWENTY STATIONS SELECTED IN STAGE ONE.                        C
C THIS PROGRAM GENERATES THE MATRIX INPUT FOR STAGE TWO OF THE C
C OPTIMIZATION.                                                 C
C INTEGER VARIABLES: 20 S02STj                                  C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC

      PARAMETER (NS=30,NF=31)
      INTEGER J,K,N
      REAL W1,W2,COEF,WCOEF1(NS,NF),WCOEF2(NF),HFDF,STAT,
     &FAIRSH

      OPEN (9,FILE='TEST.DAT',STATUS='UNKNOWN')
      OPEN (10,FILE='TOYWT.DAT',STATUS='OLD')
```

```
      OPEN (11,FILE='OBJT7.DAT',STATUS='OLD')

C     initialize variables
C     stat1 represents that the 20 stations chosen will each get
C     one bundle of HFDFs, the stat2 says that 10 of those stations
C     can receive another bundle of HFDFs

      HFDF = 240
      STAT1 = 20
      STAT2 = 10
      FAIRSH = 8

      DO 100 N=1,24

C       weightings for objective functions 1, and 2
        READ(10,700) W1,W2
C        WRITE(9,*) N,W1,W2

C       weighted coefficients for objective function 1
        DO 20 J=1,NS
          DO 30 K=1,NF
            READ(11,705) COEF
C           WRITE(9,*)J,K,COEF
            WCOEF1(J,K) = W1*COEF*1.0
C            WRITE(9,*)J,K,W1,W2,WCOEF1(J,K)
 30       CONTINUE
 20     CONTINUE

        REWIND(11)
C       weighted coefficients for excess coverage of freq k
```

```
      DO 40 K=1,NF
            WCOEF2(K) = W2 * (-1)
C            WRITE(9,*) W2,WCOEF2(K)
  40    CONTINUE

C     open file to write sas input for each weighting N
      IF (N.EQ.1) OPEN (12,FILE='[JSTEPPE.THESIS.TIME7]F7W1.SAS',
     &                  STATUS='NEW')
      IF (N.EQ.2) OPEN (12,FILE='[JSTEPPE.THESIS.TIME7]F7W2.SAS',
     &                  STATUS='NEW')
      IF (N.EQ.3) OPEN (12,FILE='[JSTEPPE.THESIS.TIME7]F7W3.SAS',
     &                  STATUS='NEW')
      IF (N.EQ.4) OPEN (12,FILE='[JSTEPPE.THESIS.TIME7]F7W4.SAS',
     &                  STATUS='NEW')
      IF (N.EQ.5) OPEN (12,FILE='[JSTEPPE.THESIS.TIME7]F7W5.SAS',
     &                  STATUS='NEW')
      IF (N.EQ.6) OPEN (12,FILE='[JSTEPPE.THESIS.TIME7]F7W6.SAS',
     &                  STATUS='NEW')
      IF (N.EQ.7) OPEN (12,FILE='[JSTEPPE.THESIS.TIME7]F7W7.SAS',
     &                  STATUS='NEW')
      IF (N.EQ.8) OPEN (12,FILE='[JSTEPPE.THESIS.TIME7]F7W8.SAS',
     &                  STATUS='NEW')
      IF (N.EQ.9) OPEN (12,FILE='[JSTEPPE.THESIS.TIME7]F7W9.SAS',
     &                  STATUS='NEW')
      IF (N.EQ.10)OPEN(12,FILE='[JSTEPPE.THESIS.TIME7]F7W10.SAS',
     &                  STATUS='NEW')
      IF (N.EQ.11)OPEN(12,FILE='[JSTEPPE.THESIS.TIME7]F7W11.SAS',
     &                  STATUS='NEW')
      IF (N.EQ.12)OPEN(12,FILE='[JSTEPPE.THESIS.TIME7]F7W12.SAS',
     &                  STATUS='NEW')
```

```
      IF (N.EQ.13)OPEN(12,FILE='[JSTEPPE.THESIS.TIME7]F7W13.SAS',
     &                   STATUS='NEW')
      IF (N.EQ.14)OPEN(12,FILE='[JSTEPPE.THESIS.TIME7]F7W14.SAS',
     &                   STATUS='NEW')
      IF (N.EQ.15)OPEN(12,FILE='[JSTEPPE.THESIS.TIME7]F7W15.SAS',
     &                   STATUS='NEW')
      IF (N.EQ.16)OPEN(12,FILE='[JSTEPPE.THESIS.TIME7]F7W16.SAS',
     &                   STATUS='NEW')
      IF (N.EQ.17)OPEN(12,FILE='[JSTEPPE.THESIS.TIME7]F7W17.SAS',
     &                   STATUS='NEW')
      IF (N.EQ.18)OPEN(12,FILE='[JSTEPPE.THESIS.TIME7]F7W18.SAS',
     &                   STATUS='NEW')
      IF (N.EQ.19)OPEN(12,FILE='[JSTEPPE.THESIS.TIME7]F7W19.SAS',
     &                   STATUS='NEW')
      IF (N.EQ.20)OPEN(12,FILE='[JSTEPPE.THESIS.TIME7]F7W20.SAS',
     &                   STATUS='NEW')
      IF (N.EQ.21)OPEN(12,FILE='[JSTEPPE.THESIS.TIME7]F7W21.SAS',
     &                   STATUS='NEW')
      IF (N.EQ.22)OPEN(12,FILE='[JSTEPPE.THESIS.TIME7]F7W22.SAS',
     &                   STATUS='NEW')
      IF (N.EQ.23)OPEN(12,FILE='[JSTEPPE.THESIS.TIME7]F7W23.SAS',
     &                   STATUS='NEW')
      IF (N.EQ.24)OPEN(12,FILE='[JSTEPPE.THESIS.TIME7]F7W24.SAS',
     &                   STATUS='NEW')
C     write the SAS input file

      WRITE (12,*) 'OPTIONS LINESIZE=78;'
      WRITE (12,*)
      WRITE (12,*) 'TITLE ''LOCATING A SAR NETWORK WITH GOOD HFDF
     & FREQUENCY ASSIGNMENTS'';'
```

```
      WRITE (12,*)
      WRITE (12,*) 'DATA SAR;'
      WRITE (12,*)
      WRITE (12,*) 'INPUT _TYPE_ $ _COL_ $ _ROW_ $
     & _COEF_;'
      WRITE (12,*)
      WRITE (12,*) 'CARDS;'
C     create the objective function

C     objective function
      WRITE (12,*) 'MAX                 .        PROFIT      .'

C     objective function two data & upperbounds for obj vars
      DO 125 K=1,NF
        IF (K .LT. 10 ) THEN
          WRITE (12,760) '.            F',K,'EX     PROFIT',
     &    WCOEF2(K)
          WRITE (12,765) 'UPPERBD    F',K,'EX      UPF',K
     &    ,'EX     12.0'
        ELSE
          WRITE (12,761) '.           F',K,'EX     PROFIT',WCOEF2(K)

          WRITE (12,766) 'UPPERBD    F',K,'EX     UPF',K
     &    ,'EX     12.0'
        ENDIF
  125 CONTINUE

C     source TWO to receiving station constraint
C     10 FIXED stations will get one more bundle of HFDF resources
      WRITE(12,*) 'EQ                .        S02ST      .'
```

```
      WRITE(12,*) '.            _RHS_        S02ST       ',STAT2

      DO 135 J=1,NS
C     fixed stations
      IF ((J .EQ. 1) .OR. (J .EQ. 8) .OR. (J .EQ. 15) .OR.
     & (J .EQ.27) .OR. (J .EQ. 28) .or.
C     other fixed stations from STAGE one optimization
     & (j.eq.2).or.(j.eq.3).or.(j.eq.25).or.(j.eq.9).or.
     & (j.eq.10).or.(j.eq.12).or.(j.eq.14).or.(j.eq.16).or.
     & (j.eq.17).or.(j.eq.18).or.(j.eq.4).or.(j.eq.21).or.
     & (j.eq.22).or.(j.eq.7).or.(j.eq.29))then
C
C     objective function one
C
        DO 115 K=1,NF
         IF ((J .LT. 10) .AND. (K .LT.10)) THEN
           WRITE (12,750) '.          ST',J,'F',K,'      PROFIT '
     &     ,WCOEF1(J,K)
           WRITE (12,755) 'UPPERBD    ST',J,'F',K,'   UPST',J,'F',K


     &     ,'     1.0'
         ELSEIF (J .LT. 10) THEN
           WRITE (12,751) '.          ST',J,'F',K,'      PROFIT '
     &     ,WCOEF1(J,K)
           WRITE (12,756) 'UPPERBD    ST',J,'F',K,'   UPST',J,'F',K
     &     ,'     1.0'
         ELSEIF (K .LT. 10) THEN
           WRITE (12,752) '.          ST',J,'F',K,'      PROFIT '
     &     ,WCOEF1(J,K)
           WRITE (12,757) 'UPPERBD    ST',J,'F',K,'   UPST',J,'F',K
```

```
     &     ,'    1.0'
         ELSE
           WRITE (12,753) '.        ST',J,'F',K,'     PROFIT '
     &     ,WCOEF1(J,K)
           WRITE (12,758) 'UPPERBD  ST',J,'F',K,'   UPST',J,'F',K
     &     ,'    1.0'
         ENDIF
  115   CONTINUE
C
C     source TWO to receiving station constraint
C     10 stations will get one more bundle of HFDF resources
        IF (J .LT. 10) THEN
          WRITE(12,770) '.         SO2ST',J,'    SO2ST         1.0'
        ELSE
          WRITE(12,771) '.        SO2ST',J,'    SO2ST         1.0'
        ENDIF

C
C     fixed stations constraint from stage one
C
          if (j .lt. 10) then
            write(12,775)'EQ             .       FIXST',J
     &      ,'    .  '
            write(12,775)'.          _RHS_       FIXST',J
     &      ,'    1.0'
            write(12,780)'.         SO1ST',J,'      FIXST',J
     &      ,'    1.0'
          else
            write(12,776)'EQ             .       FIXST',J
     &      ,'    .  '
```

```
      write(12,776)'.        _RHS_      FIXST',J
     &      ,'     1.0'
      write(12,781)'.        S01ST',J,'    FIXST',J
     &      ,'     1.0'
     endif
C
C    conservation of flow at receiving stations
C
     IF (J .LT. 10) THEN
      WRITE(12,775)'EQ              .      CONST',J
     &      ,'    .  '
      WRITE(12,775)'.          _RHS_      CONST',J
     &      ,'     1.0'
      WRITE(12,780)'.         S01ST',J,'     CONST',J
     &      ,'     9.0'
      WRITE(12,780)'.         S02ST',J,'     CONST',J
     &      ,'     8.0'
      WRITE(12,740)'BINARY    S02ST',J,'  BINARY    1'
      DO 140 K=1,NF
        IF (K .LT. 10) THEN
          WRITE (12,800)'.         ST',J,'F',K,'    CONST',J

     &        ,'    -1.0'
        ELSE
          WRITE(12,802)'.         ST',J,'F',K,'     CONST',J
     &        ,'    -1.0'
        ENDIF
140    CONTINUE
     ELSE
      WRITE(12,776)'EQ              .      CONST',J
```

```
     &      ,'   . '
            WRITE(12,776)'.        _RHS_     CONST',J
     &      ,'   1.0'
            WRITE(12,781)'.       SO1ST',J,'    CONST',J
     &      ,'   9.0'
            WRITE(12,781)'.       SO2ST',J,'    CONST',J
     &      ,'   8.0'
            WRITE(12,741)'BINARY  SO2ST',J,'  BINARY   1'
            DO 145 K=1,NF
              IF (K .LT. 10) THEN
                WRITE(12,801)'.        ST',J,'F',K,'    CONST',J
     &          ,'   -1.0'
              ELSE
                WRITE(12,803)'.        ST',J,'F',K,'    CONST',J
     &          ,'   -1.0'
              ENDIF
  145       CONTINUE
          ENDIF
C
C     conservation of flow at each frequency
C
          DO 185 K=1,NF
            IF ((J .LT. 10) .AND. (K .LT. 10)) THEN
             WRITE(12,850) '.         ST',J,'F',K,'       CONF',K

     &       ,'    1.0'
            ELSE IF (J .LT. 10) THEN
             WRITE(12,851) '.        ST',J,'F',K,'       CONF',K
     &       ,'    1.0'
            ELSE IF (K .LT. 10) THEN
```

```
          WRITE(12,852) '.        ST',J,'F',K,'       CONF',K
     &        ,'    1.0'
         ELSE
          WRITE(12,853) '.        ST',J,'F',K,'      CONF',K
     &        ,'    1.0'
        ENDIF
  185     CONTINUE

        ENDIF
  135 CONTINUE
C
C     conservation of flow at each frequency
C
      DO 180 K=1,NF
        IF (K .LT. 10) THEN
          WRITE(12,835) 'EQ              .       CONF',K,'     .'
          WRITE(12,835) '.          _RHS_       CONF',K,'   0.0'
          WRITE(12,840) '.          F',K,'EX       CONF',K,' -1.0'

          WRITE(12,840) '.          F',K,'NE       CONF',K,' -1.0'

          WRITE(12,845) 'UPPERBD    F',K,'NE     UPF',K,'NE  8.0'

        ELSE
          WRITE(12,836) 'EQ              .      CONF',K,'     .'
          WRITE(12,836) '.          _RHS_      CONF',K,'   0.0'
          WRITE(12,841) '.          F',K,'EX      CONF',K,'   -1.0'

          WRITE(12,841) '.          F',K,'NE      CONF',K,'   -1.0'
```

```
      WRITE(12,846) 'UPPERBD    F',K,'NE    UPF',K,'NE    8.0'
     ENDIF
 180 CONTINUE
C
C    conservation of flow at excess coverage
C
    WRITE(12,*) 'EQ              .        CONEX     .'
    WRITE(12,*) '.          _RHS_        CONEX   0.0'
    DO 195 K=1,NF
      IF (K .LT. 10) THEN
        WRITE(12,855) '.          F',K,'EX       CONEX     1.0'
      ELSE
        WRITE(12,856) '.          F',K,'EX       CONEX     1.0'
      ENDIF
 195 CONTINUE
    WRITE(12,*) '.          EXSI        CONEX     -1.0'
    WRITE(12,*) 'UPPERBD     EXSI       UPEXSI    120.0'
C
C    conservation of flow at nonexcess coverage
C
    WRITE(12,*) 'EQ              .        CONNE     .'
    WRITE(12,*) '.          _RHS_        CONNE     0.0'
    DO 200 K=1,NF
      IF (K .LT. 10) THEN
        WRITE(12,855) '.          F',K,'NE       CONNE     1.0'
      ELSE
        WRITE(12,856) '.          F',K,'NE       CONNE     1.0'
      ENDIF
 200 CONTINUE
```

```
      WRITE(12,*) '.              KESI        COMME     -1.0'
      WRITE(12,*) 'UPPERBD        KESI        UPKESI    240.0'
C
C     sink constraint
C
      WRITE(12,*) 'EQ             .           SINK      .'
      WRITE(12,*) '.              _RES_       SINK      240.0'
      WRITE(12,*) '.              KESI        SINK      1.0'
      WRITE(12,*) '.              EXSI        SINK      1.0'

      WRITE (12,*) '; '
      WRITE (12,*) 'PROC LP SPARSEDATA POUT=SOLUTION MAXIT1=10000
     & MAXIT2=999999 IMAXIT=99999999'
      WRITE (12,*) 'PRINTFREQ=500;'
      WRITE (12,*) 'RUN;'

      CLOSE (12)

  100 CONTINUE

  700 FORMAT (2(F7.4))
  705 FORMAT (10X,F14.7)
  730 FORMAT (1X,A28,I1,A8)
  731 FORMAT (1X,A27,I2,A8)
  735 FORMAT (1X,A15,I1,A8,I1,A6)
  736 FORMAT (1X,A15,I2,A8,I2,A6)
  740 FORMAT (1X,A15,I1,A13)
  741 FORMAT (1X,A14,I2,A13)
  750 FORMAT (1X,A13,I1,A1,I1,A13,F10.6)
  751 FORMAT (1X,A12,I1,A1,I2,A13,F10.6)
```

```
752 FORMAT (1X,A12,I2,A1,I1,A13,F10.6)
753 FORMAT (1X,A11,I2,A1,I2,A13,F10.6)
755 FORMAT (1X,A13,I1,A1,I1,A8,I1,A1,I1,A8)
756 FORMAT (1X,A12,I1,A1,I2,A7,I1,A1,I2,A8)
757 FORMAT (1X,A12,I2,A1,I1,A7,I2,A1,I1,A8)
758 FORMAT (1X,A11,I2,A1,I2,A6,I2,A1,I2,A6)
760 FORMAT (1X,A13,I1,A13,F10.6)
761 FORMAT (1X,A12,I2,A13,F10.6)
765 FORMAT (1X,A13,I1,A11,I1,A11)
766 FORMAT (1X,A12,I2,A10,I2,A11)
770 FORMAT (1X,A15,I1,A21)
771 FORMAT (1X,A14,I2,A21)
773 FORMAT (1X,A15,I1,A12,I1,A8)
774 FORMAT (1X,A14,I2,A11,I2,A8)
775 FORMAT (1X,A28,I1,A8)
776 FORMAT (1X,A27,I2,A8)
780 FORMAT (1X,A15,I1,A11,I1,A8)
781 FORMAT (1X,A14,I2,A10,I2,A8)
785 FORMAT (1X,A15,I1,A11,I1,A8)
786 FORMAT (1X,A14,I2,A10,I2,A8)
790 FORMAT (1X,A13,I1,A13,I1,A8)
791 FORMAT (1X,A12,I2,A12,I2,A8)
795 FORMAT (1X,A13,I1,A11,I1,A10)
796 FORMAT (1X,A12,I2,A10,I2,A10)
800 FORMAT (1X,A13,I1,A1,I1,A11,I1,A8)
802 FORMAT (1X,A12,I1,A1,I2,A11,I1,A8)
801 FORMAT (1X,A12,I2,A1,I1,A10,I2,A8)
803 FORMAT (1X,A11,I2,A1,I2,A10,I2,A8)
810 FORMAT (1X,A26,I1,I1,A6)
811 FORMAT (1X,A25,I2,I1,A6)
```

```
815 FORMAT (1X,A12,I1,A1,I1,I1,A10,I1,I1,A8)
816 FORMAT (1X,A10,I2,A1,I2,I1,A9,I2,I1,A8)
820 FORMAT (1X,A12,I1,I1,A1,I1,A10,I1,I1,A9)
821 FORMAT (1X,A11,I1,I1,A1,I2,A10,I1,I1,A9)
822 FORMAT (1X,A11,I2,I1,A1,I1,A9,I2,I1,A9)
823 FORMAT (1X,A10,I2,I1,A1,I2,A9,I2,I1,A9)
825 FORMAT (1X,A12,I1,I1,A1,I1,A8,I1,I1,A1,I1,A8)
826 FORMAT (1X,A11,I1,I1,A1,I2,A7,I1,I1,A1,I2,A8)
827 FORMAT (1X,A11,I2,I1,A1,I1,A7,I2,I1,A1,I1,A8)
828 FORMAT (1X,A10,I2,I1,A1,I2,A6,I2,I1,A1,I2,A8)
830 FORMAT (1X,A13,I1,A22)
831 FORMAT (1X,A12,I2,A22)
835 FORMAT (1X,A27,I1,A6)
836 FORMAT (1X,A26,I2,A6)
840 FORMAT (1X,A13,I1,A13,I1,A5)
841 FORMAT (1X,A12,I2,A12,I2,A7)
845 FORMAT (1X,A13,I1,A11,I1,A7)
846 FORMAT (1X,A12,I2,A10,I2,A9)
850 FORMAT (1X,A13,I1,A1,I1,A12,I1,A8)
851 FORMAT (1X,A12,I1,A1,I2,A11,I2,A8)
852 FORMAT (1X,A12,I2,A1,I1,A12,I1,A8)
853 FORMAT (1X,A11,I2,A1,I2,A11,I2,A8)
855 FORMAT (1X,A13,I1,A22)
856 FORMAT (1X,A12,I2,A22)
    END
```

Appendix F. *Description of Floppy Disk Files*

F.1 Floppy Disk One

Floppy disk one has three directories: FORTRAN, TOY1, and TOY2. The files in each directory are described in the next subsections.

F.1.1 FORTRAN Directory. **The files in the FORTRAN directory are also listed in the appendices. These fortran program files are:**

Append1.tex Data split program.

Append2.tex Objective function coefficients program.

Append3.tex Generation of single-stage SAS input file.

Append4.tex Generation of stage-one SAS input file.

Append5.tex Generation of stage-two SAS input file.

F.1.2 TOY1 Directory. The files in the TOY1 directory are used for the first test problem which is described in Chapter 4. This toy problem uses time block one data with stations 10, 12, 14, 21, and 30, transmitters 28 through 31, and frequencies 7 through 9. The toy problem can be modified to use the same stations, transmitters, and frequencies for any time block. The files in this directory are:

D1toy1.for This FORTRAN program strips HFDF propagation data from the large D1.dat file created by the data split program in appendix one.

D1toy1.dat This file is the HFDF propagation data created by D1toy1.for.

F1toy1.for This FORTRAN program strips frequency data from the large F1.dat file created by the data split program in appendix one.

F1toy1.dat This file is the frequency data created by D1toy1.for.

Trigtoy.for This FORTRAN program strips trig data for the test problem from Trigdata.dat and reads it into trigtoy1.dat.

Trigtoy.dat This file is the trig data file created by Trigtoy.for.

Alphatoy.for This FORTRAN program uses Trigdata.dat and calculates $I_{\alpha i}$s used in the nonlinear objective function.

Alphatoy.dat This file is the $I_{\alpha i}$s created by Alphatoy.for.

Nonlincl.for This FORTRAN program uses D1toy.dat, F1toy.dat, and Alphatoy.dat to calculate coefficients for the nonlinear objective function.

Nonlincl.dat This file is the nonlinear objective function coefficients created by Nonlincl.for.

Dat1toy1.for This FORTRAN program uses Nonlinc.dat to build the input file used by the zero-one nonlinear optimization code.

Dat1toy1.dat This file is the input data file for Zlinctoy.for created by Dat1toy1.for.

Zlinctoy.for This FORTRAN program is the zero-one nonlinear optimization code which optimizes with no initial starting conditions. It requires Dat1toy1.dat as an input file (4).

Zlinittoy.for This FORTRAN program is Zlinctoy.for which has been modified to use specific initial starting conditions. These starting conditions can be modified internally from run to run.

Ratiotoy.dat This file is the weight data W_{ij} used for the first objective function of the MOLIP.

Objftoy1.for This FORTRAN program uses the D1toy.dat, F1toy.dat, and Ratiotoy.dat to calculate coefficients for the first objective function of the MOLIP.

Objftoy1.dat This file is the objective function coefficients for the first objective function of the MOLIP.

Jtemptoy1.ifi This file is the Ifile input for ADBASE.

Jtemptoy1.qfi This file is the Qfile input for ADBASE.

F.1.3 TOY2 Directory. The files in the TOY2 directory are used for the second test problem which is described in Chapter 4. This toy problem uses time block one data with stations 10, 12, 14, 21, and 30, transmitters 21 through 24, and frequencies 7 through 9, 16 and 17. The toy problem can be modified to use the same stations, transmitters, and frequencies for any time block. The files in this directory are the same as the files in directory TOY1 except they are modified to use the specific transmitters and frequencies for this test problem.

F.2 Floppy Disk Two

The top level directory has Udfeval.for, a FORTRAN program which produces EVAL results. Udfeval.for requires an formatted input file with the tasking result to be evaluated. Formatted input files with thesis results are found in the four directories: TIME1, COVER1, TIME7, and COVER7. COVER implies that a covering constraint is used for the results in COVER directories, whereas no covering constraint is used with files in TIME directories. In general, the files in the directories are named either T*W! or T*FIX!. If the file is named T*W! the * is the time block and the ! is λ weight. For instance, !=1 corresponds to λ_1=1.0 and !=2 corresponds to λ_1=0.99. If the file is named T*FIX!, the * is the time block and the ! corresponds to the fixed weighting on the second objective function.

Appendix G. *Illustration of MOLIP versus NLIP Test Problem 1*

This appendix illustrates the specific formulation and test data used for the first test problem which compares NLIP and MOLIP formulations in Chapter 4 . The detailed results for this test case. Details are presented in Section 4.4.

G.1 *Specific Formulations*

The MOLIP and NLIP formulations are presented with summation indices that are specific to test problem 1.

G.1.1 MOLIP formulation.

$$\max \sum_{i}^{4}\sum_{j}^{5}\sum_{k}^{3} W_{ij}F_{ik}P_{ijk}X_{jk}$$

$$\min \sum_{k}^{3} Y_k$$

subject to

$$\begin{aligned} X_j &= 1, \quad \forall\, j \in \{1,2,3\} \\ \sum_{j}^{5} X_j &\leq NS \\ \sum_{j}^{5}\sum_{k}^{3} X_{jk} &\leq NH \end{aligned}$$

where NH is the number of HFDF receivers

$$\begin{aligned} X_{jk} - X_j &\leq 0, \quad \forall\, j,\, k \\ \sum_{j}^{5} X_{jk} - Y_k &\leq FS, \quad \forall\, k \\ X_j &\in \{0,1\} \\ X_{jk} &\in \{0,1\} \\ Y_k &\geq 0 \text{ and integer.} \end{aligned}$$

G.1.2 NLIP formulation.

$$\max \sum_{i}^{4}\sum_{k}^{3} F_{ik} \sum_{\alpha\in C} U_{i\alpha k}(X) I_{i\alpha}$$

where $U_{i\alpha k}(X) = \left[\prod_{j\in\alpha} P_{ijk}X_{jk}\right]\left[\prod_{h\notin\alpha}(1 - P_{ihk}X_{hk})\right]$
subject to

$$\begin{aligned} X_j &= 1, \quad \forall\ j \in \{1,2,3\} \\ \sum_{j}^{5} X_j &\leq NS \\ \sum_{j}^{5}\sum_{k}^{3} X_{jk} &\leq NH \end{aligned} \tag{16}$$

where NH is the number of HFDF receivers

$$\begin{aligned} X_{jk} - X_j &\leq 0, \quad \forall\ j,k \\ X_j &\in \{0,1\} \\ X_{jk} &\in \{0,1\}. \end{aligned} \tag{17}$$

G.2 Test Problem Data Files

The files in the TOY1 directory on floppy disk 1 are used for this test problem. This toy problem uses time block one data with stations 10, 12, 14, 21, and 30, transmitters 28 through 31, and frequencies 7 through 9. The toy problem can be modified to use the same stations, transmitters, and frequencies for any time block. Appendix 6 describes the specific files used to generate the data. Table 10 illustrates the frequency transmission data denoted by F_{ik} in the formulations. Tables 11, 12, 13, and 14 illustrate the propagation data denoted by P_{ijk} in the formulations. Table 15 illustrates the accuracy weighting

function data denoted by W_{ij}. Table 16 illustrates the confidence region indicator data denoted by $I_{\alpha i}$.

Table 10. Frequency Transmission Data (F_{ik}) for Test Case 1

i/k	freq 1	freq 2	freq 3
trans 1	0.04	0.04	0.04
trans 2	0.00	0.00	0.01
trans 3	0.03	0.05	0.05
trans 4	0.00	0.00	0.00

Table 11. Frequency Propagation Transmitter 1 Data (P_{1jk}) for Test Case 1

j/k	freq 1	freq 2	freq 3
station 1	0.98	0.95	0.96
station 2	0.98	0.98	0.98
station 3	0.90	0.92	0.83
station 4	0.97	0.98	0.90
station 5	0.98	0.94	0.94

Table 12. Frequency Propagation Transmitter 2 Data (P_{2jk}) for Test Case 1

j/k	freq 1	freq 2	freq 3
station 1	0.32	0.13	0.33
station 2	0.44	0.08	0.30
station 3	0.15	0.46	0.31
station 4	0.01	0.01	0.01
station 5	0.29	0.04	0.19

G.3 TOY1 Input Files for MOLIP

G.3.1 Jtemptoy1.ifi. This is the first of two formatted ADBASE input files for the first test problem in Chapter 4. It contains primarily the constraint and coefficient information.

Table 13. Frequency Propagation Transmitter 3 Data (P_{3jk}) for Test Case 1

j/k	freq 1	freq 2	freq 3
station 1	0.51	0.35	0.52
station 2	0.13	0.01	0.10
station 3	0.58	0.71	0.51
station 4	0.01	0.12	0.01
station 5	0.01	0.01	0.00

Table 14. Frequency Propagation Transmitter 4 Data (P_{4jk}) for Test Case 1

j/k	freq 1	freq 2	freq 3
station 1	0.01	0.01	0.01
station 2	0.01	0.01	0.01
station 3	0.01	0.01	0.01
station 4	0.01	0.01	0.01
station 5	0.01	0.01	0.01

Table 15. Accuracy-Weighting Function Data (W_{ij}) for Test Case 1

i/k	station 1	station 2	station 3	station 4	station 5
trans 1	0.3808	0.7407	0.1951	0.1210	0.7956
trans 2	0.1477	0.1301	0.1140	0.0596	0.2504
trans 3	0.1471	0.0892	0.1580	0.0834	0.1509
trans 4	0.0515	0.7679	0.0615	0.0820	0.0427

Table 16. Confidence Region Indicator Function Data ($I_{\alpha i}$) for Test Case 1

α/i	stations included	trans 1	trans 2	trans 3	trans 4
combo 1	1,2,3	1	0	0	0
combo 2	1,2,4	1	0	0	0
combo 3	1,2,5	1	1	0	0
combo 4	1,3,4	1	0	0	0
combo 5	1,3,5	1	1	1	0
combo 6	1,4,5	1	1	1	0
combo 7	2,3,4	1	0	0	0
combo 8	2,3,5	1	1	1	0
combo 9	2,4,5	1	1	0	0
combo 10	3,4,5	1	1	1	0
combo 11	1,2,3,4	1	0	1	0
combo 12	1,2,3,5	1	1	1	0
combo 13	1,2,4,5	1	1	1	0
combo 14	1,3,4,5	1	1	1	0
combo 15	2,3,4,5	1	1	1	0

```
Steppe SAR toy problem with 2 objectives and no pairwise constraint
     5001        2       23        23        2         0         0        45
       63
   1  3  1.0          1  4  1.0          1  5  1.0          2  6  1.0
   2  7  1.0          2  8  1.0          2  9  1.0          2 10  1.0
   2 11  1.0          2 12  1.0          2 13  1.0          2 14  1.0
   2 15  1.0          2 16  1.0          2 17  1.0          2 18  1.0
   2 19  1.0          2 20  1.0          3  3 -1.0          3 12  1.0
   4  3 -1.0          4 13  1.0          5  3 -1.0          5 14  1.0
   6  4 -1.0          6 15  1.0          7  4 -1.0          7 16  1.0
   8  4 -1.0          8 17  1.0          9  5 -1.0          9 18  1.0
  10  5 -1.0         10 19  1.0         11  5 -1.0         11 20  1.0
  12  6  1.0         12  9  1.0         12 12  1.0         12 15  1.0
  12 18  1.0         12 21 -1.0         13  7  1.0         13 10  1.0
  13 13  1.0         13 16  1.0         13 19  1.0         13 22 -1.0
```

```
14  8  1.0      14 11  1.0      14 14  1.0      14 17  1.0
14 20  1.0      14 23 -1.0      15  3  1.0      16  4  1.0
17  5  1.0      18  6  1.0      19  7  1.0      20  8  1.0
21  9  1.0      22 10  1.0      23 11  1.0
      14
 1      2.0      2     10.0     12      3.0     13      3.0
14      3.0     15      1.0     16      1.0     17      1.0
18      1.0     19      1.0     20      1.0     21      1.0
22      1.0     23      1.0
       2
1  1  1.0       2  2  1.0
       2
1     1.0       2     1.0
       0
       0
      18
 1  6  1.7178     1  7  1.70447    1  8  1.89347    1  9  2.93833
 1 10  2.9080     1 11  2.98717    1 12   .97728    1 13  1.27887
 1 14  1.08597    1 15   .47198    1 16   .52436    1 17   .44037
 1 18  3.12328    1 19  2.99900    1 20  3.03903    2 21 -1.0
 2 22 -1.0        2 23 -1.0
       0
```

G.3.2 Jtemptoy1.qfi. This is the second of two formatted ADBASE input files for the first test problem used in Chapter 4. This contains input and output parameters for ADBASE that are related to problem being solved.

```
***--****1*********2-----****3*********4****** ADBASE MODE = 1 SECTION
   1. NUMB            1    (NUMBER OF PROBLEMS TO BE SOLVED)
----------------------------
```

```
 2. MODE              1   (REGULAR OR RANDOM PROBLEM MODE) 1,2
----------------------------
 3. IFASE2            2   (PHASE II OPTION) 1 TO 5
 4. IFASE3            2   (PHASE III OPTION) 0,1,2
 5. IWEAK             0   (EFFICIENT OR WEAKLY-EFFICIENT) 0,1
----------------------------
 6. MLISTB        16000   (MAXIMUM NUMBER OF EFFICIENT BASES) <2500
 7. IZFMT             0   (EXPONENTIAL/FIXED FORMAT IN ZFILE) 0 TO 6
----------------------------
 8. IPRINT(1)         1   (OBVIOUS ERRORS) 0,1
 9. IPRINT(2)         0   (PROBLEM COEFFICIENTS) 0,1
10. IPRINT(3)         3   (NOTHING/BASES/EXTREME PTS) 0/1,2,3/4,5,6
11. IPRINT(4)         1   (EFFICIENCY TOTALS) 0,1
12. IPRINT(5)         0   (INDIVIDUAL PROBLEM DATA) 0,1
13. IPRINT(6)         0   (CUMULATIVE DATA) 0,1
14. IPRINT(7)         0   (CODE LISTS) 0,1
15. IPRINT(8)         1   (ZFILE) 0,1
16. IPRINT(9)         1   (REDUCED COSTS AND TABLEAUS) 0,1,2
17. IPRINT(10)        0   (LFILE) 0,1
18. IPRINT(11)        0   (PREMULTIPLICATION T-MATRIX) 0,1
----------------------------
19. IV9L              1   (BEGINNING TABLEAU VARIABLE)
20. IV9U             23   (ENDING TABLEAU VARIABLE)
21. I9L               1   (TABLEAUS BEGIN AT THIS BASIS)
22. I9U              23   (TABLEAUS END AT THIS BASIS)
23. I10L              1   (LFILE BEGINS ON WAY TO THIS BASIS)
24. I10U             23   (LFILE ENDS AT THIS BASIS)
***--****1*********2-----****3*********4*********5*** MODE = 2 SECTION
25. NSTART         5001   (STARTING PROBLEM NUMBER)
26. NOBJS             2   (NUMBER OF OBJECTIVES)
```

```
27. N1              23   (NUMBER OF STRUCTURAL VARIABLES)
28. IK              23   (LESS THAN OR EQUAL TO CONSTRAINTS)
----------------------------
29. JZDEN           25   (PERCENT A-MATRIX ZERO DENSITY) 0 TO 100
30. JLA             -1   (A-COEFFICIENT LOWER LIMIT)
31. JUA              8   (UPPER LIMIT)
32. JLB             20   (B VALUE LOWER LIMIT)
33. JUB             30   (UPPER LIMIT)
34. JLC             -3   (C-COEFFICIENT LOWER LIMIT)
35. JUC              5   (UPPER LIMIT)
36. IPRINT(12)       1   (PFILE) 0,1
----------------------------
37. KSEED         6467   (SEED TO RANDOM NUMBER GENERATOR) <99999
```

G.4 TOY1 Input and Output Files for NLIP

These files are included for illustration of typical input for test problems.

G.4.1 Dat1toy1.dat. This is the formatted input file required by the zero-one solver Zlinctoy.for and is used for the first test problem in Chapter 4.

```
 20 48 21
0 0 0 0 0 1 0 0 1 0 0 1 0 0 0 0 0 0 0 0  3.4574401E-02
0 0 0 0 0 1 0 0 1 0 0 0 0 0 1 0 0 0 0 0  3.7263501E-02
0 0 0 0 0 1 0 0 1 0 0 0 0 0 0 0 0 1 0 0  3.7647702E-02
0 0 0 0 0 1 0 0 0 0 0 1 0 0 1 0 0 0 0 0  3.4221601E-02
0 0 0 0 0 1 0 0 0 0 0 1 0 0 0 0 0 1 0 0  3.4663100E-02
0 0 0 0 0 1 0 0 0 0 0 0 0 0 1 0 0 1 0 0  3.7265100E-02
0 0 0 0 0 0 0 0 1 0 0 1 0 0 1 0 0 0 0 0  3.4221601E-02
0 0 0 0 0 0 0 0 1 0 0 1 0 0 0 0 0 1 0 0  3.4596998E-02
0 0 0 0 0 0 0 0 1 0 0 0 0 0 1 0 0 1 0 0  3.7263501E-02
```

```
0 0 0 0 0 0 0 0 0 0 0 1 0 0 1 0 0 1 0 0  3.4223299E-02
0 0 0 0 0 1 0 0 1 0 0 1 0 0 1 0 0 0 0 0 -0.1006000
0 0 0 0 0 1 0 0 1 0 0 1 0 0 0 0 0 1 0 0 -0.1016603
0 0 0 0 0 1 0 0 1 0 0 0 0 0 1 0 0 1 0 0 -0.1095548
0 0 0 0 0 1 0 0 0 0 0 1 0 0 1 0 0 1 0 0 -0.1006133
0 0 0 0 0 0 0 0 1 0 0 1 0 0 1 0 0 1 0 0 -0.1009688
0 0 0 0 0 1 0 0 1 0 0 1 0 0 1 0 0 1 0 0  0.1643320
0 0 0 0 0 0 1 0 0 1 0 0 1 0 0 0 0 0 0 0  3.4260798E-02
0 0 0 0 0 0 1 0 0 1 0 0 0 0 0 1 0 0 0 0  3.6495201E-02
0 0 0 0 0 0 1 0 0 1 0 0 0 0 0 0 0 0 1 0  3.5005599E-02
0 0 0 0 0 0 1 0 0 0 0 0 1 0 0 1 0 0 0 0  3.4260798E-02
0 0 0 0 0 0 1 0 0 0 0 0 1 0 0 0 0 0 1 0  3.2986600E-02
0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 1 0 0 1 0  3.5026599E-02
0 0 0 0 0 0 0 0 0 1 0 0 1 0 0 1 0 0 0 0  3.5342701E-02
0 0 0 0 0 0 0 0 0 1 0 0 1 0 0 0 0 0 1 0  3.3903699E-02
0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 1 0 0 1 0  3.6111001E-02
0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 1 0 0 1 0  3.3942800E-02
0 0 0 0 0 0 1 0 0 1 0 0 1 0 0 1 0 0 0 0 -0.1007118
0 0 0 0 0 0 1 0 0 1 0 0 1 0 0 0 0 0 1 0 -9.6616700E-02
0 0 0 0 0 0 1 0 0 1 0 0 0 0 0 1 0 0 1 0 -0.1029165
0 0 0 0 0 0 1 0 0 0 0 0 1 0 0 1 0 0 1 0 -9.6645303E-02
0 0 0 0 0 0 0 0 0 1 0 0 1 0 0 1 0 0 1 0 -9.8650701E-02
0 0 0 0 0 0 1 0 0 1 0 0 1 0 0 1 0 0 1 0  0.1578051
0 0 0 0 0 0 0 1 0 0 1 0 0 1 0 0 0 0 0 0  3.1234600E-02
0 0 0 0 0 0 0 1 0 0 1 0 0 0 0 0 1 0 0 0  3.3868801E-02
0 0 0 0 0 0 0 1 0 0 1 0 0 0 0 0 0 0 0 1  3.5562199E-02
0 0 0 0 0 0 0 1 0 0 0 0 0 1 0 0 1 0 0 0  2.8684800E-02
0 0 0 0 0 0 0 1 0 0 0 0 0 1 0 0 0 0 0 1  3.0153999E-02
0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 1 0 0 1  3.2492701E-02
0 0 0 0 0 0 0 0 0 0 1 0 0 1 0 0 1 0 0 0  2.9282400E-02
```

```
0 0 0 0 0 0 0 0 0 0 1 0 0 1 0 0 0 0 0 1  3.0760501E-02
0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 1 0 0 1  3.3168901E-02
0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 1 0 0 1  2.8093100E-02
0 0 0 0 0 0 0 1 0 0 1 0 0 1 0 0 1 0 0 0 -8.4320001E-02
0 0 0 0 0 0 0 1 0 0 1 0 0 1 0 0 0 0 0 1 -8.8196103E-02
0 0 0 0 0 0 0 1 0 0 1 0 0 0 0 0 1 0 0 1 -9.5513798E-02
0 0 0 0 0 0 0 1 0 0 0 0 0 1 0 0 1 0 0 1 -8.0834999E-02
0 0 0 0 0 0 0 0 0 0 1 0 0 1 0 0 1 0 0 1 -8.4471501E-02
0 0 0 0 0 0 0 1 0 0 1 0 0 1 0 0 1 0 0 1  0.1321234
  0.0000000E+00  0.0000000E+00  0.0000000E+00  0.0000000E+00  0.0000000E+00
   1.000000       1.000000       1.000000       1.000000       1.000000
   1.000000       1.000000       1.000000       1.000000       1.000000
   1.000000       1.000000       1.000000       1.000000       1.000000      10
  0.0000000E+00  0.0000000E+00   1.000000       1.000000       1.000000
  0.0000000E+00  0.0000000E+00  0.0000000E+00  0.0000000E+00  0.0000000E+00
  0.0000000E+00  0.0000000E+00  0.0000000E+00  0.0000000E+00  0.0000000E+00
  0.0000000E+00  0.0000000E+00  0.0000000E+00  0.0000000E+00  0.0000000E+00 2
   1.000000      0.0000000E+00  0.0000000E+00  0.0000000E+00  0.0000000E+00
  0.0000000E+00  0.0000000E+00  0.0000000E+00  0.0000000E+00  0.0000000E+00
  0.0000000E+00  0.0000000E+00  0.0000000E+00  0.0000000E+00  0.0000000E+00
  0.0000000E+00  0.0000000E+00  0.0000000E+00  0.0000000E+00  0.0000000E+00 1
  0.0000000E+00   1.000000      0.0000000E+00  0.0000000E+00  0.0000000E+00
  0.0000000E+00  0.0000000E+00  0.0000000E+00  0.0000000E+00  0.0000000E+00
  0.0000000E+00  0.0000000E+00  0.0000000E+00  0.0000000E+00  0.0000000E+00
  0.0000000E+00  0.0000000E+00  0.0000000E+00  0.0000000E+00  0.0000000E+00 1
  -1.000000      0.0000000E+00  0.0000000E+00  0.0000000E+00  0.0000000E+00
  0.0000000E+00  0.0000000E+00  0.0000000E+00  0.0000000E+00  0.0000000E+00
  0.0000000E+00  0.0000000E+00  0.0000000E+00  0.0000000E+00  0.0000000E+00
  0.0000000E+00  0.0000000E+00  0.0000000E+00  0.0000000E+00  0.0000000E+00 -1
  0.0000000E+00  -1.000000      0.0000000E+00  0.0000000E+00  0.0000000E+00
```

0.0000000E+00	0.0000000E+00	0.0000000E+00	0.0000000E+00	0.0000000E+00	
0.0000000E+00	0.0000000E+00	0.0000000E+00	0.0000000E+00	0.0000000E+00	
0.0000000E+00	0.0000000E+00	0.0000000E+00	0.0000000E+00	0.0000000E+00	-1
-1.000000	0.0000000E+00	0.0000000E+00	0.0000000E+00	0.0000000E+00	
1.000000	0.0000000E+00	0.0000000E+00	0.0000000E+00	0.0000000E+00	
0.0000000E+00	0.0000000E+00	0.0000000E+00	0.0000000E+00	0.0000000E+00	
0.0000000E+00	0.0000000E+00	0.0000000E+00	0.0000000E+00	0.0000000E+00	0
-1.000000	0.0000000E+00	0.0000000E+00	0.0000000E+00	0.0000000E+00	
0.0000000E+00	1.000000	0.0000000E+00	0.0000000E+00	0.0000000E+00	
0.0000000E+00	0.0000000E+00	0.0000000E+00	0.0000000E+00	0.0000000E+00	
0.0000000E+00	0.0000000E+00	0.0000000E+00	0.0000000E+00	0.0000000E+00	0
-1.000000	0.0000000E+00	0.0000000E+00	0.0000000E+00	0.0000000E+00	
0.0000000E+00	0.0000000E+00	1.000000	0.0000000E+00	0.0000000E+00	
0.0000000E+00	0.0000000E+00	0.0000000E+00	0.0000000E+00	0.0000000E+00	
0.0000000E+00	0.0000000E+00	0.0000000E+00	0.0000000E+00	0.0000000E+00	0
0.0000000E+00	-1.000000	0.0000000E+00	0.0000000E+00	0.0000000E+00	
0.0000000E+00	0.0000000E+00	0.0000000E+00	1.000000	0.0000000E+00	
0.0000000E+00	0.0000000E+00	0.0000000E+00	0.0000000E+00	0.0000000E+00	
0.0000000E+00	0.0000000E+00	0.0000000E+00	0.0000000E+00	0.0000000E+00	0
0.0000000E+00	-1.000000	0.0000000E+00	0.0000000E+00	0.0000000E+00	
0.0000000E+00	0.0000000E+00	0.0000000E+00	0.0000000E+00	1.000000	
0.0000000E+00	0.0000000E+00	0.0000000E+00	0.0000000E+00	0.0000000E+00	
0.0000000E+00	0.0000000E+00	0.0000000E+00	0.0000000E+00	0.0000000E+00	0
0.0000000E+00	-1.000000	0.0000000E+00	0.0000000E+00	0.0000000E+00	
0.0000000E+00	0.0000000E+00	0.0000000E+00	0.0000000E+00	0.0000000E+00	
1.000000	0.0000000E+00	0.0000000E+00	0.0000000E+00	0.0000000E+00	
0.0000000E+00	0.0000000E+00	0.0000000E+00	0.0000000E+00	0.0000000E+00	0
0.0000000E+00	0.0000000E+00	-1.000000	0.0000000E+00	0.0000000E+00	
0.0000000E+00	0.0000000E+00	0.0000000E+00	0.0000000E+00	0.0000000E+00	
0.0000000E+00	1.000000	0.0000000E+00	0.0000000E+00	0.0000000E+00	

0.0000000E+00	0.0000000E+00	0.0000000E+00	0.0000000E+00	0.0000000E+00	0
0.0000000E+00	0.0000000E+00	-1.000000	0.0000000E+00	0.0000000E+00	
0.0000000E+00	0.0000000E+00	0.0000000E+00	0.0000000E+00	0.0000000E+00	
0.0000000E+00	0.0000000E+00	1.000000	0.0000000E+00	0.0000000E+00	
0.0000000E+00	0.0000000E+00	0.0000000E+00	0.0000000E+00	0.0000000E+00	0
0.0000000E+00	0.0000000E+00	-1.000000	0.0000000E+00	0.0000000E+00	
0.0000000E+00	0.0000000E+00	0.0000000E+00	0.0000000E+00	0.0000000E+00	
0.0000000E+00	0.0000000E+00	0.0000000E+00	1.000000	0.0000000E+00	
0.0000000E+00	0.0000000E+00	0.0000000E+00	0.0000000E+00	0.0000000E+00	0
0.0000000E+00	0.0000000E+00	0.0000000E+00	-1.000000	0.0000000E+00	
0.0000000E+00	0.0000000E+00	0.0000000E+00	0.0000000E+00	0.0000000E+00	
0.0000000E+00	0.0000000E+00	0.0000000E+00	0.0000000E+00	1.000000	
0.0000000E+00	0.0000000E+00	0.0000000E+00	0.0000000E+00	0.0000000E+00	0
0.0000000E+00	0.0000000E+00	0.0000000E+00	-1.000000	0.0000000E+00	
0.0000000E+00	0.0000000E+00	0.0000000E+00	0.0000000E+00	0.0000000E+00	
0.0000000E+00	0.0000000E+00	0.0000000E+00	0.0000000E+00	0.0000000E+00	
1.000000	0.0000000E+00	0.0000000E+00	0.0000000E+00	0.0000000E+00	0
0.0000000E+00	0.0000000E+00	0.0000000E+00	-1.000000	0.0000000E+00	
0.0000000E+00	0.0000000E+00	0.0000000E+00	0.0000000E+00	0.0000000E+00	
0.0000000E+00	0.0000000E+00	0.0000000E+00	0.0000000E+00	0.0000000E+00	
0.0000000E+00	1.000000	0.0000000E+00	0.0000000E+00	0.0000000E+00	0
0.0000000E+00	0.0000000E+00	0.0000000E+00	0.0000000E+00	-1.000000	
0.0000000E+00	0.0000000E+00	0.0000000E+00	0.0000000E+00	0.0000000E+00	
0.0000000E+00	0.0000000E+00	0.0000000E+00	0.0000000E+00	0.0000000E+00	
0.0000000E+00	0.0000000E+00	1.000000	0.0000000E+00	0.0000000E+00	0
0.0000000E+00	0.0000000E+00	0.0000000E+00	0.0000000E+00	-1.000000	
0.0000000E+00	0.0000000E+00	0.0000000E+00	0.0000000E+00	0.0000000E+00	
0.0000000E+00	0.0000000E+00	0.0000000E+00	0.0000000E+00	0.0000000E+00	
0.0000000E+00	0.0000000E+00	0.0000000E+00	1.000000	0.0000000E+00	0
0.0000000E+00	0.0000000E+00	0.0000000E+00	0.0000000E+00	-1.000000	

```
0.0000000E+00  0.0000000E+00  0.0000000E+00  0.0000000E+00  0.0000000E+00
0.0000000E+00  0.0000000E+00  0.0000000E+00  0.0000000E+00  0.0000000E+00
0.0000000E+00  0.0000000E+00  0.0000000E+00  0.0000000E+00    1.000000      0
```

G.4.2 Zeroout1.dat. This is the output file from Zlinctoy.for for the first test problem in Chapter 4.

```
FEASIBLE INCUMBANT SOLUTION #           1
OBJECTIVE VALUE OF INCUMBENT SOLUTION =   0.1051765
VARIABLE            1 = 1
VARIABLE            2 = 1
VARIABLE            3 = 1
VARIABLE            4 = 1
VARIABLE            6 = 1
VARIABLE            7 = 1
VARIABLE            8 = 1
VARIABLE            9 = 1
VARIABLE           10 = 1
VARIABLE           11 = 1
VARIABLE           12 = 1
VARIABLE           13 = 1
VARIABLE           14 = 1
VARIABLE           15 = 1
FEASIBLE INCUMBANT SOLUTION #           2
OBJECTIVE VALUE OF INCUMBENT SOLUTION =   0.1054567
VARIABLE            1 = 1
VARIABLE            2 = 1
VARIABLE            3 = 1
VARIABLE            4 = 1
VARIABLE            6 = 1
```

VARIABLE 7 = 1

VARIABLE 8 = 1

VARIABLE 9 = 1

VARIABLE 10 = 1

VARIABLE 11 = 1

VARIABLE 12 = 1

VARIABLE 13 = 1

VARIABLE 14 = 1

VARIABLE 16 = 1

FEASIBLE INCUMBANT SOLUTION # 3

OBJECTIVE VALUE OF INCUMBENT SOLUTION = 0.1075858

VARIABLE 1 = 1

VARIABLE 2 = 1

VARIABLE 3 = 1

VARIABLE 4 = 1

VARIABLE 6 = 1

VARIABLE 7 = 1

VARIABLE 8 = 1

VARIABLE 9 = 1

VARIABLE 10 = 1

VARIABLE 11 = 1

VARIABLE 12 = 1

VARIABLE 13 = 1

VARIABLE 14 = 1

VARIABLE 17 = 1

FEASIBLE INCUMBANT SOLUTION # 4

OBJECTIVE VALUE OF INCUMBENT SOLUTION = 0.1078107

VARIABLE 1 = 1

VARIABLE 2 = 1

VARIABLE 3 = 1

```
VARIABLE            4 = 1
VARIABLE            6 = 1
VARIABLE            7 = 1
VARIABLE            8 = 1
VARIABLE            9 = 1
VARIABLE           10 = 1
VARIABLE           11 = 1
VARIABLE           12 = 1
VARIABLE           13 = 1
VARIABLE           15 = 1
VARIABLE           17 = 1
FEASIBLE INCUMBANT SOLUTION #          5
OBJECTIVE VALUE OF INCUMBENT SOLUTION =   0.1080909
VARIABLE            1 = 1
VARIABLE            2 = 1
VARIABLE            3 = 1
VARIABLE            4 = 1
VARIABLE            6 = 1
VARIABLE            7 = 1
VARIABLE            8 = 1
VARIABLE            9 = 1
VARIABLE           10 = 1
VARIABLE           11 = 1
VARIABLE           12 = 1
VARIABLE           13 = 1
VARIABLE           16 = 1
VARIABLE           17 = 1
FEASIBLE INCUMBANT SOLUTION #          6
OBJECTIVE VALUE OF INCUMBENT SOLUTION =   0.1098202
VARIABLE            1 = 1
```

```
VARIABLE          2 = 1
VARIABLE          3 = 1
VARIABLE          4 = 1
VARIABLE          6 = 1
VARIABLE          7 = 1
VARIABLE          8 = 1
VARIABLE          9 = 1
VARIABLE         10 = 1
VARIABLE         11 = 1
VARIABLE         12 = 1
VARIABLE         14 = 1
VARIABLE         16 = 1
VARIABLE         17 = 1
FEASIBLE INCUMBANT SOLUTION #           7
OBJECTIVE VALUE OF INCUMBENT SOLUTION =   0.1100451
VARIABLE          1 = 1
VARIABLE          2 = 1
VARIABLE          3 = 1
VARIABLE          4 = 1
VARIABLE          6 = 1
VARIABLE          7 = 1
VARIABLE          8 = 1
VARIABLE          9 = 1
VARIABLE         10 = 1
VARIABLE         11 = 1
VARIABLE         12 = 1
VARIABLE         15 = 1
VARIABLE         16 = 1
VARIABLE         17 = 1
FEASIBLE INCUMBANT SOLUTION #           8
```

OBJECTIVE VALUE OF INCUMBENT SOLUTION = 0.1102749

VARIABLE 1 = 1

VARIABLE 2 = 1

VARIABLE 3 = 1

VARIABLE 4 = 1

VARIABLE 6 = 1

VARIABLE 7 = 1

VARIABLE 8 = 1

VARIABLE 9 = 1

VARIABLE 10 = 1

VARIABLE 11 = 1

VARIABLE 13 = 1

VARIABLE 14 = 1

VARIABLE 15 = 1

VARIABLE 17 = 1

FEASIBLE INCUMBANT SOLUTION # 9

OBJECTIVE VALUE OF INCUMBENT SOLUTION = 0.1107800

VARIABLE 1 = 1

VARIABLE 2 = 1

VARIABLE 3 = 1

VARIABLE 4 = 1

VARIABLE 6 = 1

VARIABLE 7 = 1

VARIABLE 8 = 1

VARIABLE 9 = 1

VARIABLE 10 = 1

VARIABLE 11 = 1

VARIABLE 13 = 1

VARIABLE 15 = 1

VARIABLE 16 = 1

VARIABLE 17 = 1

FEASIBLE INCUMBANT SOLUTION # 10

OBJECTIVE VALUE OF INCUMBENT SOLUTION = 0.1125093

VARIABLE 1 = 1

VARIABLE 2 = 1

VARIABLE 3 = 1

VARIABLE 4 = 1

VARIABLE 6 = 1

VARIABLE 7 = 1

VARIABLE 8 = 1

VARIABLE 9 = 1

VARIABLE 10 = 1

VARIABLE 11 = 1

VARIABLE 14 = 1

VARIABLE 15 = 1

VARIABLE 16 = 1

VARIABLE 17 = 1

FEASIBLE INCUMBANT SOLUTION # 11

OBJECTIVE VALUE OF INCUMBENT SOLUTION = 0.1127499

VARIABLE 1 = 1

VARIABLE 2 = 1

VARIABLE 3 = 1

VARIABLE 5 = 1

VARIABLE 6 = 1

VARIABLE 7 = 1

VARIABLE 8 = 1

VARIABLE 9 = 1

VARIABLE 10 = 1

VARIABLE 11 = 1

VARIABLE 13 = 1

VARIABLE 18 = 1

VARIABLE 19 = 1

VARIABLE 20 = 1

FEASIBLE INCUMBANT SOLUTION # 12

OBJECTIVE VALUE OF INCUMBENT SOLUTION = 0.1133375

VARIABLE 1 = 1

VARIABLE 2 = 1

VARIABLE 4 = 1

VARIABLE 5 = 1

VARIABLE 6 = 1

VARIABLE 7 = 1

VARIABLE 8 = 1

VARIABLE 9 = 1

VARIABLE 10 = 1

VARIABLE 11 = 1

VARIABLE 15 = 1

VARIABLE 16 = 1

VARIABLE 17 = 1

VARIABLE 20 = 1

FEASIBLE INCUMBANT SOLUTION # 13

OBJECTIVE VALUE OF INCUMBENT SOLUTION = 0.1137217

VARIABLE 1 = 1

VARIABLE 2 = 1

VARIABLE 4 = 1

VARIABLE 5 = 1

VARIABLE 6 = 1

VARIABLE 7 = 1

VARIABLE 8 = 1

VARIABLE 9 = 1

VARIABLE 10 = 1

VARIABLE 11 = 1

VARIABLE 16 = 1

VARIABLE 17 = 1

VARIABLE 18 = 1

VARIABLE 20 = 1

SOLUTION OPTIMAL

1.000000	1.000000	0.0000000E+00	1.000000	1.000000
1.000000	1.000000	1.000000	1.000000	1.000000
1.000000	0.0000000E+00	0.0000000E+00	0.0000000E+00	0.0000000E+00
1.000000	1.000000	1.000000	0.0000000E+00	1.000000

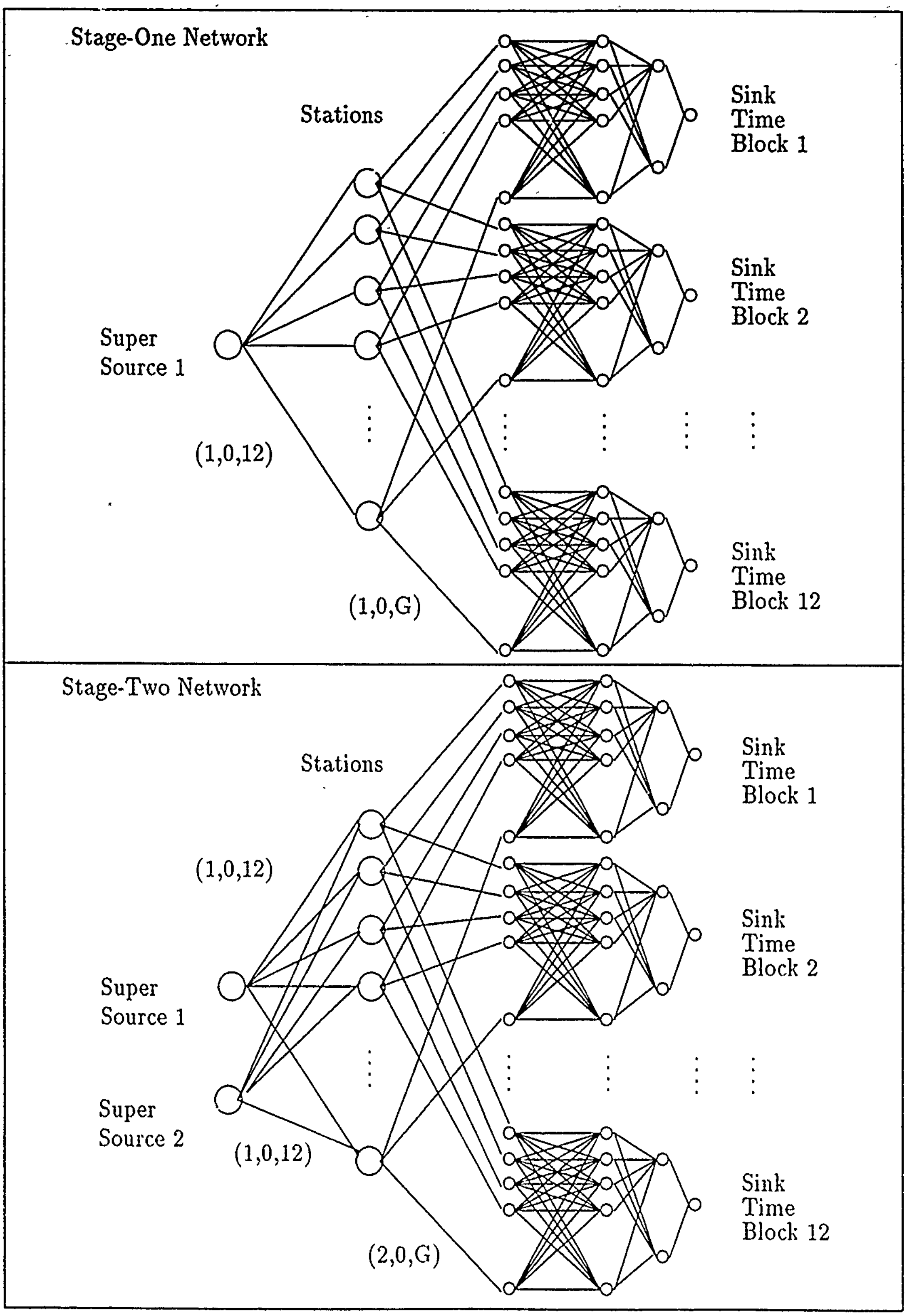
OPTIMAL VALUE = 0.1137217

NUMBER OF FEASIBLE INCUMBENT SOLUTIONS = 13

Appendix H. *Multi-time Period Concept of the Two-Stage MOLIP*

Figure 16 illustrates how the two-stage MOLIP can be expanded to simultaneously encompass 12 time periods for locating HFDFs in a SAR network. Only one set of arcs needs to be integerized: the arcs emanating from super source 1 in stage one, an l the arcs emanating from super source 2 in stage two. Stations and bundles which are used by each time block are located by first set of arcs. Application of this concept to the multi-time period GSARP is a logical extension for future research as discussed in Section 7.5.

Figure 16. Multi-time Period Two-stage MOLIP Concept



Bibliography

1. Batta, Rajan, J.M. Dolan, and N. N. Krishnamurthy. "The Maximal Expected Covering Location Problem: Revisisted," *Transportation Science 23*:277-287 (November 1989).

2. Berman, Oded, and Edward H. Kaplan. "Equity Maximizing Facility Location Schemes," *Transportation Science 24*:137-144 (May 1990).

3. Brandeau, Margaret L. and Samuel S. Chiu. "An Overview of Representative Problems in Location Research," *Management Science 35*:645-674 (June 1989).

4. Chrissis, James W. "The solution of Nonlinear Pseudo-Boolean Optimization Problems Subject to Linear Constraints." PhD Dissertation, Department of Industrial Engineering and Operations Research, VPI, Blacksburg, VA, 1980.

5. Cohen, Sara W., David A. Drake, and Alfred B. Marsh. Conversation at National Security Agency, 28 June 1990.

6. Cohen, Sara W., David A. Drake, and Alfred B. Marsh. Conversation at National Security Agency, 29 September 1990.

7. Daskin, Mark S. "A Maximum Expected Covering Location Model: Formulation, Properties and Heuristic Solution," *Transportation Science 17*:48-69 (February 1983).

8. DoD. "EVAL." An Evaluation program provided on floppy for evaluation of thesis computations, November 1990.

9. DoD. "Unclassified High Frequency Direction Finding (HFDF) Problem Data," Description provided in writing, 1 SEP 90.

10. Drake David A. Telephone Interview, 0830, 12 Feb 1991.

11. Drake, Capt David A. and Alfred B. Marsh. "High Frequency Direction Finding Problem Statement," Distributed to students and faculty in the Department of Operational Sciences, Air Force Institute of Technology (AU), Wright-Patterson AFB OH, Spring 1989.

12. H.E. Daniels. "The Theory of Position Finding," *Journal of the Royal Statistical Society*, 13:2, 1951.

13. Hogan, Kathleen and Charles Revelle. " Concepts and Applications of Backup Coverage," *Management Science 32*:1434-1444 (November 1986).

14. Johnson, Krista E. *Frequency Assignments for HFDF Receivers in a Search and Rescue Network*. Master Thesis. School of Engineering, Air Force Institute of Technology (AU), Wright-Patterson AFB OH, March, 1990.

15. Johnson, Krista, Yupo Chan, Alfred B. Marsh, and James Chrissis. "Frequency Assignment and Direction Finder Allocation in a Generalized Search and Rescue Location Problem" Working paper. School of Engineering, Air Force Institute of Technology (AU), Wright-Patterson AFB OH, May, 1990.

16. Louveaux, Francois S. and Jacques Francois Thisse. "Production and Location on a Network under Demand Uncertainty," *Operations Research Letters 4*:145-149 (December 1985).

17. Marsh, Alfred B., Capt David A. Drake, and Sara W. Cohen. "High Frequency Direction Finding (HFDF) Facility Location Problem Statement," Distributed to students and faculty in the Department of Operational Sciences. Air Force Institute of Technology (AU), Wright-Patterson AFB OH, Spring 1990.

18. Minoux, Michel. *Mathematical Programming Theory and Algorithms*. New York: John Wiley and Sons, Inc., 1986.

19. Nemhauser, George L., and Laurence A. Wolsey. *Integer and Combinatorial Optimization*. New York: John Wiley and Sons, Inc., 1988.

20. Olson, David L. and Anne Dillinger. "The Impact of Missing Objectives," Department of Business Analysis, Texas A&M University, College Station, TX 77843.

21. Pirkul, Hasan and David Schilling. "The Capacitated Maximal Covering Location Problem with Backup Service," *Annals of Operations Research 18*:141-154 (February 1989).

22. Steuer, Ralph E. (1989). "ADBASE Multiple Objective Linear Programming Package," Department of Management Science and Information Technology, University of Georgia, Athens, Georgia, USA.

23. Steuer, Ralph E. *Multiple Criteria Optimization: Theory, Computation, and Application*. New York: John Wiley and Sons, Inc., 1986.

Vita

Captain Jean M. Steppe was born on 7 April 1963 in Bangor, Maine. In 1981, she graduated from Beaver High School in Beaver, PA and attended the United States Air Force Academy. In 1986, she graduated from the Academy with a Bachelor of Science degree in Operations Research. Her first assignment was as a scientific analyst for the 4486th Fighter Weapons Squadron at Eglin Air Force Base, Florida. Her primary duties were as a supporting analyst for the Reconnaissance Evaluation Program and the Weapon Systems Evaluation Program. She entered the School of Engineering, Air Force Institute of Technology in August of 1989.

Permanent address: Box 357
Beaver, PA 15009

REPORT DOCUMENTATION PAGE

Form Approved
OMB No 0704-0188

1. AGENCY USE ONLY (Leave blank)	2. REPORT DATE March 1991	3. REPORT TYPE AND DATES COVERED Master's Thesis
4. TITLE AND SUBTITLE LOCATING DIRECTION FINDERS IN A GENERALIZED SEARCH AND RESCUE NETWORK		5. FUNDING NUMBERS
6. AUTHOR(S) Jean M. Steppe, Captain, USAF		
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) Air Force Institute of Technology, WPAFB OH 45433-6583		8. PERFORMING ORGANIZATION REPORT NUMBER AFIT/GOR/ENS/91M-17
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES) Department of Defense Attn: R06 9800 Savage Road Ft George G Meade MD 20755		10. SPONSORING/MONITORING AGENCY REPORT NUMBER
11. SUPPLEMENTARY NOTES		
12a. DISTRIBUTION/AVAILABILITY STATEMENT Approved for public release; distribution unlimited		12b. DISTRIBUTION CODE

13. ABSTRACT (Maximum 200 words)

A multiobjective linear programming approach is applied to the problem of locating receiving stations and HFDF receivers in a search and rescue network in order to maximize the expected number of distress signals that are geolocated. The multiobjective formulation is made up of two contrasting objectives: one maximizes the expected accurate lines of bearing, and one minimizes the excess coverage in the network. The individual objectives are weighted and combined into a composite objective function. The resulting problem is expressed as a two-stage network flow problem and is solved with SAS LP using a limited number of binary variables. The problem is solved iteratively for several weightings of the composite objective function. Solutions are evaluated by a FORTRAN program provided by the Department of Defense. In all cases, the best results were three to four standard deviations better than a sample of 1000 or more heuristically tasked random network configurations. These results demonstrate that a two-stage multiobjective formulation consistently provides good feasible network configurations and is therefore a practical alternative to the robust, yet intractable, nonlinear integer formulation.

14. SUBJECT TERMS Mathematical Programming; Multiobjective Optimization, Integer Programming, Network Flow; Search and Rescue; Direction Finders; HFDF			15. NUMBER OF PAGES 166
			16. PRICE CODE
17. SECURITY CLASSIFICATION OF REPORT Unclassified	18. SECURITY CLASSIFICATION OF THIS PAGE Unclassified	19. SECURITY CLASSIFICATION OF ABSTRACT Unclassified	20. LIMITATION OF ABSTRACT UL

NSN 7540-01-280-5500 Standard Form 298